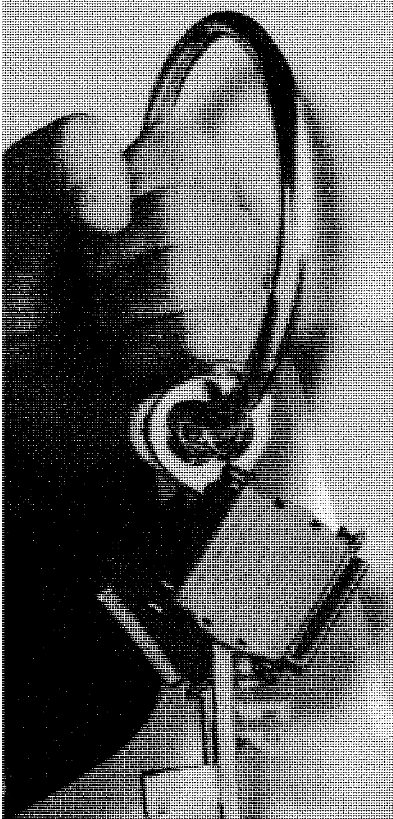
حماية و فك حماية البرامج

منظومة - MS-DOS



مجموعـة من الباحثيـن

حماية ونزع حماية البرامج

«منظومة MS DOS»

ترجمة : م. فايز البرشــة

مراجعة: د.م. مؤنس كيالي

الطبعة الأولى ـ 1997

دار الطليعة الجديدة

سوريا ـ دمشق ـ ص.ب 34494

هـ: 7775872

اخراج: هـالـة فطوم

صمم الغلاف: جمال سعيد

تنفيذ الغلاف: بلاتينيوم

مقدمة

ما يزال الاهتمام بأمن وسرية المعلومات ومعالجتها في أنظمة الحاسب محصوراً حتى الآن في مجموعة محدودة من الاختصاصيين في بلادنا، وهذا الاهتمام مطروح بدرجات مختلفة من الجدية من خلال مجموعة من الأجهزة الحكومية والعسكرية وبعض الدوائر العلمية.

إن تحول بلادنا نحو اقتصاد السوق سيجلب معه حتماً ظهور عدد كبير من المؤسسات والمصارف التي ترتبط فعالياتها باستخدام الحواسب الألكترونية، كما تدل على ذلك أيضاً تجربة البلدان الأخرى. وعندما يعي مسؤولو تلك المؤسسات ذلك ستكون أمامهم مهمة حماية ما لديهم من معلومات سرية حدّية. ويقصد هنا بالمعلومات الحدية تلك المعلومات ذات الدرجات المختلفة للسرية : معلومات للاستخدام الخاص في العمل، معلومات تشكل سراً تجارياً، أو أسرار الشركة، المعلومات التي تعتبر ملكاً خاصاً لبعض المؤسسات أو الأشخاص.. الخ.

من المفيد جداً التفكير بالتجربة الأجنبية التي بحوزتنا، فمازال هناك وقت كاف للتفكير، كي لا نعيد بطريقتنا اختراع الدراجة الهوائية مجدداً، فمن جهة تقوم مؤسسات ومنظمات ومصممون خاصون بإنتاج وسائل حماية منظومات الحاسب وخاصة المعلومات، لأنهم يعانون من عوائق محسوسة في مجال تصريف منتجاتهم لعدم وجود سوق ثابتة ومستقرة، ونقص الدعية وسبكة الخدمات... الخ.

ومن جهة ثانية فإن عدداً متزايداً من المنشآت التجارية، والمصارف، والبورصة، والتعاونيات، والمنشآت الصغيرة.. الخ، التي تستخدم أنظمة الحاسب تعاني من حاجة ملحة لحماية المعلومات التي لديها أو تعتقد

بضرورة تلك الحماية، لكنها غير قادرة على تقييم احتياجاتها في هذا المجال، ولا تدري إلى من يجب التوجه في سبيل المساعدة المتخصصة، للوصول إلى وسائل حماية أنظمة الحاسب.

كما أن الكثيرين من مقتني أنظمة الحاسب ليس لديهم تصور حول جوهر المسألة والتقنيات المستخدمة لحلها اليوم، وهذا ليس ذنبهم بل مشكلتهم.

لا يوجد تفسير مباشر لهذا التناقض لأنه عملياً كانت هناك عوامل كثيرة قد لعبت دورها في تكوين الوضع الحالي المؤسف، ويمكن القول وبثقة كبيرة بأن أحد العوامل السلبية لذلك يكمن في عدم كمال قوانيننا الناظمة في مجال معالجة المعلومات. بل أنه في الواقع ليس لدينا قانون حول المعلومات، إذا استثنينا التعليمات الإدارية والأحكام الواردة فيها حول حماية الأسرار الحكومية والعسكرية.

ولكن ماذا عن الأسرار التجارية؟ وماذا عن المعلومات السرية الشخصية والخاصة؟ وماذا عن حقوق الملكية الفكرية؟

توجد هياكل حكومية قوية بما فيها المحاكم قادرة على حماية الأسرار الحكومية والعسكرية. ولكن من يحمي برامجنا من النسخ ومنتجاتنا من القرصنة؟! ومن سيتحمل مسؤولية إشهار المعلومات السرية المتعلقة بنشاط المنشآت غير الحكومية؟ ومن سيعوض للمتضررين من إشهار تلك المعلومات؟ وهنا قد يتبادر لذهن الكثيرين سؤال فحواه: أية خسائر؟

ولا يعني غياب التجربة والخبرة في تحديد الخسائر أنها غير موجود في العالم.

ووفقاً لتقديرات الاختصاصيين النمساويين على سبيل المثال فإنه خلال عام ١٩٨٨ تحمّل الإقتصاد النمساوي خسارة مقدارها (١،٥ مليار شيلينغ) بسبب تقطع وانحراف عمل البرامج. ووفقاً لمعطيات المركز القومي الأمريكي للمعلومات فإن جرائم الحاسب في نفس السنة الحقت بالشركات الأمريكية خسارة تقدر بنحو (٥٠٠ مليون دولار). وخاصة المصارف، وحسب تصريح «هوارد غلاسمان» المسؤول الأمني «لبنك أمريكا» فإن جرائم وأخطاء الحاسب يمكن أن تصبح سبباً لانهيار المنظومة الاقتصادية للبلد.

الخطر على أمن المعلومات يأتي من جهتين:

يأتي الخطر على منظومة الحاسب وما فيها من معلومات هي قيد المعالجة، من جانب «هواة» المعلومات المهتمين أساساً بمنظومات الحاسب وليس بما تحوية من معلومات.

وضمن هذه الفئة يأتي مستخدمو الكمبيوتر الهواة وخاصة من لديهم معلومات جزئية حول الحواسب العصرية وهم معنيون باظهور وكأنهم مثقفون في هذا المجال، وتجب الإشارة لعامل آخر يساعد هؤلاء الهواة ويكمن في إتاحة فرصة حرة في تشريعات الدول الصناعية يسميه بعض الأخصائيين «مجال الحاسب».

وأحد أُسس ذلك التشريع يقوم في تحديد مبدأ سرية المعلومات وذلك انطلاقاً من مواصفات المعلومات ذاتها وليس كما هو قائم لدينا من اعتبارات تجعل أحياناً من ورقة بيضاء تماماً «سراً» فقط مروسة بكلمة «سري»، ولنلاحظ بأن الخبرة التشريعية لا تقف مكانها، وهي تتطور وتتكامل باستمرار. ومع نشوء أنواع جديدة من جرائم الكمبيوتر تظر تشريعات جديدة مناسبة. وهكذا سن الكونغرس الأمريكي عام ١٩٨٦ قانوناً حول الاحتيال وسوء الاستعمال المترافق مع استخدام وسائط الحاسب وبمساعدتها.

ووفقاً لأحكام هذا القانون فإن محاولة التسلل غير المشروعة لمنظومة الحاسب الإلكترونية والتي لا يعود الحاسب بنتيجتها قادراً على القيام بوظائفه.

كما يعتبر القانون تغيير أو تخريب المعلومات بالطرق غير المشروعة جرماً يعاقب عليه بالسجن حتى عشر سنوات. وفي عدد من البلدان ووفقاً للقوانين النافذة يمكن أن يتعرض للمسؤلية إلى جانب المخالف والمرتكب للجرم أيضاً «ضحايا» جرائم الحاسب أنفسهم.

مثلاً الشركة التي تقدم خدمات معلوماتية ولديها معلومات سريّة عن المتعاملين معها، يمكن أن تتهم في حال كشف تسرب معلوماتها بعدم اتخاذ الإجراءات المناسبة للحماية من هذا التسرب، وقد تتعرض للمطالبة

بالتعويض المادي ولغرامات كبيرة، كما أن شركات التأمين قد تمتنع عن دفع التعويضات المناسبة للشركة عن خسارتها بفقدان المعلومات. وعلى الرغم من وجود التشريعات الناظمة لذلك في الولايات المتحدة لم يتوقف مغامرو الحاسب عن أفعالهم.

وهكذا اخترق المدعو «غولديس» باعتباره «لص الحاسب» المحترف، فقد اخترق وهاجم منظومات الحاسب في العديد من المصارف وشركات التأمين وبعض المؤسسات الصناعية بهدف كشف نقاط الضعف في أنظمة حمايتها والتي تسهل للعاملين إمكانية سرقة أو تشويه المعطيات.

ويعتبر «غولديس» أستاذ مبرمج في دائرة المنظومة العملياتية **MYS** وهي منظومة كبيرة ومعقدة من انتاج شركة **IBM**، وتجد خدماته إضافة لعدد من الأخصائيين في هذا الحقل طلباً متنامياً، فقد استطاع خلال الأعوام الثلاثة السابقة أن يتسلل لأنظمة (٢٥) شركة عملاقة. واليوم أصبح الكل قادر على القيام بعمليات بسيطة للتقليد والنسخ غير المعتمد وغير المشروع من خلال أجهزة الحاسب الإلكتروني من طراز **IBM PC**. والانتشار الواسع والعاصف لهذه العملية في بلادنا ليس فقط بسبب ظهور وسائل حسابية ملائمة وإمكانية غالبية المستخدمين على التقليد والنسخ مع أمر **(COPY)** بل بالعوامل الأساسية التالية أيضاً:

- ضعف القوانين المرتبطة بمسائل حقوق المؤلف في المنتج البرمجي.
- وجود السياسة الحكومية المشروعة وغير المشروعة للتقليد والنسخ لتكنيك الحاسب الإلكتروني والضمان البرامجي وهو ما أصبح تقليداً منذ عدة سنوات.
- عدم إمكانية الإمتلاك العلني والنظامي لغالبية منتجات البرامج لعدم توفر القطع الأجنبي لدى المبرمجين والمستخدمين.
- غياب علاقات السوق «الطبيعية» في بلادنا.
- الرغبة في استعراض الإمكانيات والقدرات الذهنية.
- السعي نحو الثراء.

إن قضية حماية الضمانات البرمجية من التقليد والنسخ غير المشروع قائمة في ذات جوهر الإنسان السيكولوجية، وستبقى كذلك ما دام المنتج البرمجي يعتبر بضاعة، وحتى ذلك الوقت توجد ضرورة لحماية المنتج البرمجي. فسوف يظهر دائماً باحثون عن وسائل لنزع حماية البرامج وهو ما يسمسه «هواة المعلومات» ب «تعرية البرنامج». كما يوجد سعي لإيجاد آلية حماية عامة شاملة وآلية مضادة لإيجاد نازع حماية عام وشامل، حتى وقت قريب كان المبرمجون المحترفون يرون أنه من غير الملائم القيام بحماية برنامج خاص. ولذلك كان نازعو الحماية مهرة، فالقائمون بعملية النسخ غير المشروع لا يأخذون معهم مواد صالحة للاستخدام البشري (مثل لصوص المنازل).

وان عملية النسخ لا تنقص من البرنامج شيء بل تتم وهو قائم وموجود على عكس سرقة تلفاز من منزل مثلاً، وتتحول العملية تدريجياً إلى لعبة ذكاء لا نهائية نحو الكمال.

ويمكن تقسيم وسائل الحماية والضمان البرمجي إلى: وسائل حقوقية، إدارية، تنظيمية، إدارية، اقتصادية، جازية، وبرمجية.

وسائل الحماية الحقوقية البرامجية في الدول الغربية معروفة بشكل مفصل ويمكن تلخيصها كما يلي:

توجد في الدول الرأسمالية المتقدمة تشريعات وتوجد قضايا. وفي بلادنا لا توجد تشريعات مناسبة متعلقة بأمور الحاسب وبرامجه ولا قضايا تقريباً، والشاهد غير المباشر على ذلك يكمن في توجيه السؤال التالي لستخدمي الحاسب الشخصي: «أية برامج من التي بحوزتكم تم شراؤها؟!»

غالباً سيكون الجواب هو الصمت.

لن نتطرق هنا لأجراءات الحماية الإدارية – التنظيمية – وتلك الإجراءات التي تقوم بها الشركات المصنعة لمتابعة الاقتناء العلني والشرعي للبرامج. وهي تعمل لتسجيل المستخدمين المقتنين لمنتجات البرمجة وتتخذ الإجراءات الموازية كي يصبح النسخ غير المشروع غير مفيد للقائمين به.

ويمكن استخدام الإجراءات التالية لحصر وتسجيل المستخدمين:

- تلقي الوثائق والمجلات المتخصصة بشكل دوري.
- الإعلام الدائم والمنتظم لتعليم بالمتغيرات والأخطاء المكتشفة.
- إقامة الدورات لتعليم استخدام مُنتج البرامج وعقد اللقاءات العلمية.
- توفير إمكانية الحصول على استشارة علمية من المصادر المختصة.
- تقديم الحسومات عند الشراء.

يجب التنبه لأجهزة الحماية وذكرها فقط بمقدار ارتباطها مع وسائل البرمجة.

الفصل الأول

مسائل حماية معلومات الحاسب وطرائق حلها

1.1 أسس أمان منظومات الحاسب:

قبل النظر بأسس وقواعد الضمان في معالجة المعلومات في منظومة الحاسب، يجب تحديد ما نفهمه من عنوان «منظومة الحاسب».

نسمي «منظومة حاسبية» ذلك المجموع للأجهزة ووسائل البرمجة ومختلف أنواع النواقل الفيزيائية للمعلومات وخاصة المعطيات. ويشمل ذلك العاملين والقائمين على خدمة هذه العناصر. ويعتبر الأخصائيون مسألة ضمان حماية منظومة الحاسب أكثر المسائل أهمية وتعقيداً في مجال المعالجة المؤتمتة للمعلومات. إن خسائر منظومة الحاسب مفهوم واسع فعلاً. ولا يعتبر خسارة الضرر الظاهر في أحد العناصر المؤلفة للمنظومة فقط بل الوصول أيضاً بعناصر المنظومة لوضع تصبح فيه غير قادرة على العمل وهناك قضايا معقدة فمختلف أنواع تسرب المعلومات (مثلاً النسخ غير المشروع للبرامج والحصول على المعطيات السرية..الخ) وكذلك تغيير بعض الخصائص والسمات الفيزيائية والمنطقية لمنظومة الحاسب (مثلاً.. إضافة غير واردة سابقاً لنص في منظومة الملفات، زيادة حمل المنظومة من خلال إدخال عملية إضافية غير ملحوظة.. الخ) وتحديد الخسارة الممكنة لمنظومة الحاسب هي قضايا معقدة تماماً ومرتبطة بالعديد من الشروط. مثلاً هل تعتبر الملكية الفكرية في هذا البلد أو ذاك أمراً قانونياً أم لا؟ وهل يسمح بما هو متعارف عليه Copyright وهل تنظر المحاكم في التعويض عن الخسائر المعنوية التي تحيق ببعض الأشخاص. تلك الأسئلة تبين وبوضوح كيف أن

بلادنا تدخل عالم الكمبيوتر المعاصر بشكل غير كافٍ من الاستعداد والتجهيز.

ومن هنا فإن مالكي أجهزة الحاسب في غالبيتهم يعتقدون أن اقتناء برنامج لقاء نقود تدفع للمصمم أو ممثله أمراً غريباً وغير مفهوم طالما توجد الإمكانية لنسخ وتوزيع هذا البرنامج ذاته وبصورة نصف علنية ولكن مجاناً من خلال نسخه ممن يوجد لديه وأحياناً دون أذنه.

يمكن القول وبثقة بأنه حيث يتم استخدام الحاسب توجد أخطار كامنة في إلحاق الضرر المباشر أو غير المباشر للمالكين القانونين والمستخدمين الشرعيين لهذه الحواسب، وفي هذا المجال فالسؤال عن قيمة المعلومة بحد ذاتها يستحق الاهتمام.

من المتعارف عليه ومن خلال الممارسة العملية عالمياً في مجال الحاسب فإن ثمن المعلومة يعادل مقدار الخسارة الناجمة عن فقدانها مع الأخذ بعين الإعتبار النفقات المترتبة إعادة تجديدها. يمكن تقسيم مسائل أمن منظومات الحاسب وشروطها إلى المجموعات التالية:

١. مسائل ضمان الأمان الفيزيائي للعناصر المؤلفة لمنظومة الحاسب وهنا يمكن تعداد قضايا حماية المنظومة من الحريق، الفيضانات، الكوارث الطبيعية، انقطاع التغذية، السرقة، الضرر...الخ.

٢. مسائل ضمان الأمان المنطقي لعناصر المنظومة : الحماية من الاستعمال غير المؤهل والكفء، الحماية من الأخطاء المقصودة وغير المقصودة للعاملين في البرامج التي قد تلحق الضرر... الخ.

٣. قضايا ضمان وتأمين الحماية الإجتماعية لعناصر المنظومة، وهذا مرتبط بإيجاد تشريعات وقوانين ناظمة لإستعمال الأجهزة والتحقيق والمتابعة والعقوبات المترتبة جراء خرق أمان منظومات الحاسب. قد يظر ذلك وكأنه ليس هاماً، ولكن كثيراً من الأخصائيين يرون بأن إيجاد نوع من النظام في عملية الاستخدام أمر مفيد ويلعب دوراً هاماً، وكذلك صياغة بعض القواعد الأخلاقية تكون ملزمة لجميع من يعمل مع الحاسب.

وقد قام بعض خبراء الصندوق العلمي القومي الأمريكي بمحاولة لوضع ما يسمى «قواعد السلوك» موجهة لأخصائي الحاسب، حيث اعتبر أي عمل أو فكرة، خارج قواعد التهذيب فيما إذا أدت إلى :

- الإخلال بالعمل الطبيعي لمنظومات الحاسب.
- التسبب في نفقات إضافية.
- الإخلال بمصالح المستخدمين القانونية.
- التسبب في نفقات إضافية للقيام بتفتيش إضافي، إعادة بعث القدرة على العمل، وفي النهاية تحطيم أمان المنظومة... الخ.

ومن الناحية النظرية توجد أربع أصناف من المخاطر التي قد تحيق بمنظومة الحاسب:

أولاً: الانقطاع

وفي هذه الحالة من الانقطاع أو التوقف فإنه قد يكون ناتج عن فقدان عنصر من المجموعة (نتيجة السرقة مثلاً) وهذا يحصل عن طريق:

١. الاستيلاء: بعض الجهات غير المأذون لها يتاح لها الوصول للعنصر، وكمثال النسخ غير المشروع للبرامج والمعطيات والقراءة غير المأذونة للمعطيات من خلال خطوط اتصال شبكة الحاسب... الخ.

٢. التكييف، التعديل: بعض الجهات غير المأذونة لا تحصل على إمكانية الدخول للعنصر فقط بل تتحكم أيضاً. ويعتبر تعديلاً كل تغيير غير مأذون به في قاعدة المعطيات أو بشكل عام في ملفات منظومة الحاسب، كتبديل اللوغاريتم المستخدم في البرنامج من أجل القيام بعملية معالجة إضافية غير قانونية. ويتم الكشف عن التعديل أحياناً مباشرة وبسرعة؛ والتعديل الدقيق لا يمكن كشفه لوقت طويل من الزمن.

٣. التزوير: قد يضيف مغتصب البرنامج بعض العمليات الوهمية الكاذبة الضرورية له وغير الملحوظة في المنظومة: أوامر، نصوص كاذبة، في الملفات أو مستخدمين آخرين.

مثلاً بمعرفة مقاييس التسجيل في ملفات الحاسب التي يتم على أساسها احتساب الأجور في هذه المؤسسة أو تلك، يمكن تماماً أن يُدخل في هذا الملف تسجيل مزور وعند معالجة وحساب الأجور تتم أيضاً معالجة هذا التسجيل على نفس الدرجة مع التسجيل الأصلي النظامي. ولكن من سيحصل النقود؟ وهل ستحول لمستخدم غير موجود؟ حلوا اللغز بأنفسكم.

قد يقال : وماذا هناك إنها أربعة أنواع للمخاطر؟ لو تعلمون فقط كم من التحولات والأشكال المتعددة يمكن أن تظهرها هذه المخاطر الأربعة... مرة أخرى ومن حيث النظرية فإن أمان أي عنصر للمنظومة يتكون من ضمان وتأمين مواصفات ثلاث: السرية، سهولة الوصول إليه، شموليته وكماله.

● تقوم السرية في أن عنصر المنظومة يمكّن الشخص المأذون له من أن يصل اليه كمؤلف المنظومة ومستخدم البرنامج.. الخ، أما بالنسبة للآخرين فإن هذا العنصر غير موجود لأنه غير متاح.

● الشمولية والكمال بالنسبة للعنصر، تعني أنه لا يمكن لغير المؤلف والمأذون له أن يجري تعديلاً أو تبديلاً في العنصر وهي عبارة عن ضمانة صمته وسلامة العنصر في أي لحظة زمنية (القدرة على العمل). ونشير هنا بأن المقصود بالتبديل والتعديل هو عملية التسجيل والتجديد وتغيير الحالة والعزل والإنشاء. إمكانية الوصول تفترض عملياً سهولة الوصول للعنصر من الشخص المسموح له بذلك أي أن الشخص المأذون له يمكنه في أي لحظة ودون عراقيل أن يصل بسهولة للعنصر الضروري.

1.2 الحماية من النسخ غير المشروع والتقليد غير المسموح به:

معظم المصممين السوفييت للضمانات البرامجية يحاولون تحجيم النسخ والتقليد غير العلني لمنتجاتهم. عملياً طيف المنظومات المستخدمة للحماية كبير جداً، وهي غالباً يمكن نزعها من خلال تغيير (١÷٢) بايت (مثلاً يكفي غالباً تغيير الانتقال المشروط إلى اللامشروط أو العكس) وحتى المنتجات عالية التخصص والكفاءة المنتجة بمشاركة أخصائيين كبار، إلى جانب لوغاريتمات المصطلحات المعقدة يتم عند وضع أنظمة الحماية

استخدام مصائد ذكية، ميكرو مخططات خاصة، المبادلة والبديلة لـ BIOS القياسية والمطابقة للمقاييس، بمختلف «الصناديق السوداء» الموصولة مع الميناء (اللوحة) أو المركبة على الكوابل والديسكيت المفتاحية.. الخ. كل هذه الأنظمة للحماية لها نقطة ضعف تسمح بتجنب دهاء صانعيها. ويعتبر المكان الأضعف الأول هو لحظة اتخاذ القرار حول مدى علنية ومشروعية النسخة المطروحة فإذا وصلتم بمساعدة المنظم حتى هنا فهذا يعني أن الحماية قد هزمت.

في وقت ما نزع أحد الأخصائيين الحماية الأجنبية والتي أُعدت على أساس «الصندوق الأسود» والموصولة مع اللوحة، لم يكن لديه «الصندوق» الضروري المناسب، وبعد متاة طويلة ومن خلال لوغاريتم معقّد جداً ومسبب للتيه فقد وجد بأن ما تحت البرنامج المرفق في الحزمة بشكل علني يمكنه إعادة (1) أو (0) وفقاً لمنطلق التدقيق والمراقبة. وهكذا تمّ عزل كامل الحماية بعد أن أُجبر ما تحت البرنامج المكتشف على العودة دائماً للـ (0).

وللحقيقة فإنه توجد إجراءات دفاعية جيدة ضد مثل هذه الطريقة لنزع الحماية: الترميز والاصطلاحِ المعقد لنظام الحماية ذاته. وبهذه الحالة سيكون الوضع صعب جداً عند محاولته اختيار وتحديد الملف المنفذ وسيجابه بمصائد تذهب بالعقل. ويعاني نظام الحماية من ضعف آخر أكثر خطورة، سنتاوله بالتفصيل. كما تمت ملاحظته سابقاً، يجب تقبل منظومة الحاسب بشكل متكامل. ومثل هذا المدخل يسمح بالهجوم جيداً والدفاع جيداً، ولنحاول أن نقوم بذلك:

كل البرامج وبشكل عام تقوم ومن حيث الجوهر بعمل مماثل، نحصل على المعطيات من تدفق الدخل تتم معالجتها والمعلومات المعالجة توجه نحو المخرج، وأنظمة الحماية ليست استثناء. في البداية تجمع المعلومات عن المنظومة ومن ثم تعالجها (تتم عملية مشابهة للحاسب، والديسكات المفتاحية، تحدد الإطلاق والتشغيل) وفي النهاية تتخذ القرار حول الإجراءات اللاحقة.

وهكذا يظهر بأن عزل وضم نظام الحماية يمكن أن يتم في ثلاثة أمكنة:

١. في مرحلة جمع المعلومات.

٢. في مرحلة المعالجة.

٣. في مرحلة اتخاذ القرار حول الانتقال لعملية لاحقة.

وكما رأينا سابقاً فإن طرق التخريب السائرة في الطريق الثاني والثالث مرتبطة بعمل يدوي صعب لتحليل شيفرة الجماية المنزوعة. يعتبر العزل في مرحلة جمع المعلومات الأكثر إمتاعاً وأهمية وهو أكثر الطرق استقلالية وعمومية. وبالفعل لاتوجد أي ضرورة للتحليل التفصيلي لعمل منظومة الحماية المنزوعة، ويكفي محدودية تيار المعلومات التي على أساسها تتخذ الحماية قرارها وتحفظه في حالة عمل البرنامج بشكل طبيعي، بعد ذلك يجب أن نقوم بتحميل الأوامر والتي ستبدأ بتقديم النسخ عن طريق بدائل لتيارات المعلومات الداخلة.

لننظر الأمثلة التالية :

المثال الأول: وهو مكرس لنزع الحماية مع استخدام «ديسك» مفتاحي.

توجد منظومات جيدة تشكل الديسكات المفتاحية فيها صعوبة فائقة لا يمكن أخذ نسخة عنها (ولإيجاد ذلك يجب امتلاك معلومات قيمة جداً ومعدات غالية جداً لا يمكن استخدامها للنسخ المتواصل وهي شبه نادرة في بلادنا) ونورد فيما يلي طريقة تجنب مثل هذه الحماية:

١. يكتب «دريفر» خاص وهو بديل كامل لإنقطاعات الديسك.

٢. يتم تشغيل النسخ المشروع إذا كان «الدريفر» معبأ ومحمل.

٣. يحفظ ما تم الحصول عليه.

٤. يتشكل «دريفر» جديد يضم إنقطاعات الديسك.

والآن وبعد كل طلب من الديسك المفتاحي فإن «الدرايفر» يرجع المعلومات المحفوظة وليست الحقيقية. ويجب ملاحظة أن كافة العمليات المشار إليها يمكن أتمتتها بسهولة بكتابة برنامج خاص لها يحفظها الملف. كل المحاكمات الواردة أعلاه تحافظ على موثوقيتها طالما أن الحماية

تحصل على المعلومات كاملة. من خلال الانقطاع يمكن خداع الغاصب العامل بتلك الطريقة وبشكل بسيط جداً، يجب أن تبرمج بنفسك مراقبي الأجهزة الخارجية من خلال اللوحة، ويساعدنا في ذلك برنامج هائل.

طرق نزع الحماية الذكية:

١. يكتب البرنامج الذي ينقل processor إلى النظام ويشغل النّسخ المشروع للبرنامج.

وفي هذه الحالة فإن البرنامج المبحوث عنه يمنع من:

* كل عمليات الإدخال والإخراج من خلال الميناء.

* الوصول للذاكرة والتي يمكن أن تستخدم لاحتياجات المنظومة للحماية (هنا يمكن استخدام المحصلة التدقيقية، بحال معطيات BIOS الخ..)

٢. إذا كانت منظومة الحماية تحاول أن تمتلك وتستخدم نفس الأوامر ومجالات الذاكرة الممنوع الوصول إليها، فإنه من خلال آلية معالجة الحالات الخاصة، فمن السهولة كشف تلك المحاولات ومن ثم يتم رفع بروتوكول تبادل المعلومات.

يُكتب الحمل الذي يشغل النسخ غير المشروع في النظام، وتتم وبالكامل عملية الحصول على المعلومات، لدرجة أنه يتكون لدى منظومة الحماية وَهْم العمل الروتيني والنظامي. ولكن يمكن كتابة نصف آلية وهذا يعطي الإمكانية وبطريقة مجربة لاختيار المجموعة الأدنى للحدود الموضوعة على الحماية المنزوعة.

1.3 الحماية المضادة للفيروسات:

تعمل معظم المنظومات المضادة للفيروسات بصورة معيارية، فهي إما أن تقوم بتفتيش البرنامج المنفذ لتبيان وجود العدوى من عدمها وذلك من خلال مؤشرات معينة وبشكل مباشر أو غير مباشر. علامات العدوى المباشرة هي وجودها في البرنامج، الديسك، أو في منطقة محددة في الديسك.. الخ. وغير المباشرة، تغير طول الملف، حاصل الجمع التدقيقي.. الخ.

أو العدوى المتواجدة بصورة دائمة في ذاكرة الآلة، والتي تتابع مراقبة النقاط الحيوية الهامة للمنظومة العملياتية. تعتبر كل أنظمة الحماية من الفيروسات المشابهة خاماً وذلك لأن (MS DOS) ذاتها غير محمية بشكل كامل. ولذلك يمكن هنا فقط الحديث عن محاولات ترقيع مختلفة للثقوب وليس عن حماية منظومة متواصلة ومدروسة جيداً. والوضع يتغير تماماً إذا بدأنا باستخدام القسم العلوي من DOS والعاملة في (Protected mode).

وهذه البرامج توزع مستويات الامتيازات في تشغيل برامج المستخدمين بشكل أنه حتى بوجود فيروسات فيها لا يمكنها شل المنظومة العملياتية، وسيقطع الطريق على أية محاولة شريرة وبمساعدة آلية معالجة الحالات الحدية (I 286 processor) والأقدم منه، حيث تتم عملية تقييم محاولة «التغلغل» أو تنفيذ أمر ممنوع وكأنه حالة حدية تتطلب المعالج المناسب. وهكذا فالمستخدم مرة أخرى يظهر أنه محمي جيداً. ولكن «المرة الأخرى هذه» ملعونة..!

إن نظام حماية processor قائم على أساس أن لكل Segment ذاكرة بمستوى من الامتيازات، إذا كان البرنامج المتوضع في Segment الأقل امتيازاً، يحاول أن يقترب من Segment الأكثر امتيازاً – وهو ما يلاحظه processor بسهولة ويقوم بتشكيل حالة حدية – فإن تطابق كونترول المستويات والامتيازات يتحقق بشكل آلي كاملاً. أما المعلومات عن مستويات امتياز Segment فهي موجودة في اللائحة العامة والشاملة (GDT) والتي يحفظ عنوانها ريجستر (GDT). وتنتظر كامل منظومة الحماية لـ processor هذه المعلومة بفارغ الصبر على أساسها توجد إمكانية لمنع البرامج ذات الامتيازات الأدنى والأقل من تغيير المحتوى في ريجستر (GDT).

من خلال إعطاء Segment من (GDT) امتيازات أعلى يمكن سد الطريق على محاولة تغييرها غير المشروعة. وهكذا تحديداً تسلك المنظومات العملياتية العاملة مع نظام حماية فهي تعطي Segment الخاص بها أعلى مستوى من الامتيازات ولباقي العمليات المتولدة أكثر انخفاضاً.

كيف يمكن قطع مثل هذه الحماية ؟

واضح بأنه من الممكن تجنب الإيـذاء فقـط مـن خـلال تغيير مستويات الامتيازات في (GDT) والقيام بذلك سهل جداً ، ويمكـن استخدام طريقة الوصول المباشر للذاكرة لأجـل ذلك، وهي طريقة تراقب دائماً من قبل processor، عملياً في جميع منظومـات الحاسوب يوجـد «مراقبـون، مفتشون» للوصول المباشر للذاكرة والتي لا يمكن دونها تصور الأقراص الصلبة وبقية التجهيزات، ويمكن استخدام «المفتشين» أيضاً لتحطيم نظام الحماية والأمان، وهكذا فإن طريقة قطع حلقة الحماية لنظام فيرتوال تعمل كما يلي: (وبداية لنتصور أنفسنا فيروس يحاول نقل العدوى للحاسوب فماذا نفعل؟).

١. ندقق ونتحقق في النظام الذي تم إطلاقنا به، إذا كنا في النظام العملي الواقعي فلا توجد عوائق، أما إذا كنا في (نظام فيرتوال) فعلينا (تجنبه).

٢. ندرس عنوان (GDT)، وقد يكون خيراً أن مصممي (I 286) لم يلحظوا إمكانية منع عمليات قراءة (L GDT) Register)، مع أنهم قاموا بها في ريجستر (S GDT).

٣. ينبغي في Segment الخاص بنا أن يحوي مكاناً اضافياً لنسخ (GDT).

٤. ندقق مراقب الوصول المباشر للذاكرة لإعطاء المعلومات من GDT إلى مكان «البوفر» الخاص بنا.

٥. نجبر المراقب أن ينسخ عدد البايت الضروري لنا.

٦. ندرس بإنتباه نسخة (GDT) التي حصلنا عليها ونجد فيها Segment خاصتنا.

٧. نعطي Segment الإمتياز الأعلى.

٨. ننسخ ونعيد (GDT) إلى مكانه الإعتيادي باستخدام الوصول المباشر للذاكرة.

٩. والآن نفعل ما نريد.

المبرمج الماهر سيلاحظ مباشرة بأن الطريقة المشار إليها، يمكن حصارها من خلال منع البرامج المستخدمة من تنفيذ أوامر (in) و(DUT). القراءة والتسجيل مقابل الإدخال والإخراج والتي بدونها لايمكن توجيه مراقب الوصول المباشر للذاكرة. وفي هذه الحالة فإن البرمجة على الحاسب الشخصي تفقد كل بريقها وتتحول لعمل ممل أحادي الطابع، ذلك لأن المبرمج لايستطيع أن يبرمج حتى EGA/VGA وهذا يعني أنه يجب أن تقارن. ولكن حتى المقارنة الدورية للمنظومة العملياتية الجارية لـ (GDT) مع نسختها الإحتياط يمكن أن تتحول لعمل معاكس. في حال اكتشاف اختلاف، يجب تقصي أسبابه، ولا يجوز إجراء هذه المقارنة بصورة متكررة لأنها تؤدي لفقدان محسوس وذي قيمة للإنتاجية، وهذا يعني وجود نوافذ زمنية لهجوم الفيروس، والمخرج الأساسي هنا يكمن في أنه يمكن وخلال زمن لحظي وخلافاً لجميع الدفاعات تحطيم منظومة حلقة الحماية للمعالج I 286.

فما هو إذاً الطريق لمكافحة الهجوم ؟

لا تستخدموا برامج مسروقة!

للأسف فإن هذه النصيحة البسيطة عملياً لاتنفذ في ظروفنا وليس لدى الجميع دولارات فائضة لشراء الضروري له. أما ما يباع لقاء العملة المحلية فهو إما مرتفع السعر لدرجة كبيرة بالنسبة للمستخدم العادي أو أنه ليس المطلوب تماماً. وهكذا تتكون وضعية يمكن فقط للمؤسسات الثرية من خلالها أن تقتني ما تريده. وقادة هذه المؤسسات يرون أنه يجب شراء كل شيء مفيد إلا البرامج، طالما يمكن نسخها ونقلها، إذاً لدينا مخرج واحد يكمن في تشكيل وتكوين سوق لضمانات البرامج تدريجياً.

وفي حال وجود هذا السوق وبمساعدة الحماية القانونية الكاملة لحقوق المؤلف يمكن أن يزداد عدد النسخ من البرامج وبالتالي تنخفض الأسعار وهذا السوق الى الآن لم يتشكل بعد، ولذلك يقع علينا - نحن واجب الحماية من اللصوص أي أن «إنقاذ الغريق - قضية الغريق ذاته».

وتبرز مهمة حماية البرامج من النسخ غير المشروع بشكل جدي أمام المصممين.

الفصل الثاني

البنية العامة للآليات الحمائية

يخضع إجمالي منظومة الإجـراءات الحمائيـة وكذلـك عناصرهـا المكونـة الإفرادية للإمتحـان والتجريـب في الطبيعـة في مختلـف مسـتويات التطـور. فكيف يتم ذلك؟

تتم حماية منظومة العناصر متبادلة التأثـير وضمـان عـدم نزعهـا بفضـل وجود تأثير متبادل متنوع مع العناصر الخارجية في المسـتويات الفيزيائيـة، الكيميائية، البيولوجية. والوظائف الطبيعية للمنظومة المنزوعة من وسـطها يمكن الوصول إليها من خلال النقل وبدقة – ومعها – أيضاً القوى المحيطة والمتبادلة التأثير معها خارجياً. والبرنامج المغروس في غـير وسـطه الطبيعـي يجب أن يُلفظ كما يلفظ الجسم النسج الغريبة المزروعة فيه. وفي هذه الحالة فإن عملية الإنعكاس يمكن أن تستدعي تحـولات وتغـيرات غـير عكوسـة في ذاته «جهاز» الحاسوب. ولأجل تحقيق آليـة الإنعكـاس فمـن الضـروري أن يمتلك البرنـامج مبدئيـاً إمكانيـة «للتعـود» علـى وسـطه وأن يـؤدي التأثـير المتبادل له مع الوسط الأجنبي عنه لهلاكه بذاته لحظيـاً وللوسـط المحيـط. وإلا فإنه سيحمل على نفسه آثار التفكك التدريجي.

آلية الدفاع عن الحماية البرنامجية يمكن أن تضم وكمـا توحـي تجربـة العمل في هذا المجال العناصر التالية:

- قسم الحماية من نازعي الحماية «المجرد».
- قسم وحدة سمات الوسط.
- قسم مقارنة سمات الوسط.
- قسم العمليات الجوابية.

2.1 الحماية من نازع الحماية:

نازع الحماية أو المجرّد هو عبارة عن مجموعة الوسائل البرنامجية الموجهة نحو المتابعة والبحث في المنتج البرامجي المحمي، وقد لايكون هذا المصطلح علمياً تماماً ولكنه يتناسب مع ما هو قائم، والمهمة الرئيسية لقسم الحماية هذا، تقوم في تأمين حماية موثوقة من المحبين المحترفين «لتجريد» برامج الآخرين أي نزع حمايتها. وتورد مجموعة من الطرق الموجهة لتمييز التدخلات الخارجية والقضاء عليها وهي غير الملحوظة في البرنامج المحمي والواسمة عملياً لوسائل نزع الحماية.

تستخدم لنزع الحماية توليفة من طريقتين رئيسيتين: الستاتيكية والديناميكية. تلحظ الطرائق الستاتيكية تحليل نصوص البرامج المحمية في شكلها الطبيعي أوالمحور، أما الطرائق الديناميكية فتلحظ المتابعة خلف البرنامج وبمساعدة وسائط خاصة. ومعالجة نتائج العملية قد يكون مؤتمتاً جزئياً أو بالكامل.

2.1.1 مواجهة الطرائق الستاتيكية:

١. الترميز: إن التبديل والترميز في الكود البرنامجي، طريقة نموذجية لإدخال خصائص في الوسط البرنامجي، من المفيد جداً تغيير طريقة الترميز في سياق عمل البرنامج، وكمثال على ذلك نورد برنامج (CRYPT , EXE) الذي يستخدم اللوغاريتم الإتحادي الأمريكي (US Federal DES).

وبشكل عام فإن موضوع الترميز ونزعه غير واضح الأبعاد بعد، ويمكن إحالة المهتمين به إلى الإصدارات الخاصة مثلاً (10 , 9) .

٢. الضم لكتلة البرنامج ووفقاً للتغيرات الديناميكية الطارئة على العناوين المحتواة والتوقعات الأوامر الذاتية أيضاً، انتقالات محددة مثلاً الأوامر الحاصلة بمساعدة الجمع والطرح.

٣. الإنتقال الخفي: عملياً يتم استخدام العودة من ما تحت البرنامج (RET) عوضاً عن أوامر الإنتقال المباشر غير المشروط (JMB)، ومبدئياً يسجل في Stack عنوان الإنتقال الذي يتبدل في مجرى عمل البرنامج

مباشرة إلى Stack عنـوان الإنتقـال الـذي يتبـدل في مجـرى عمـل البرنـامج مباشرة إلى Stack ذاته.

٤. طريقة «بـابي – يـاغي»: وهنـا يتجـدد ويتحـدد مباشـرة في مجـال الأوامر المنفذة، مما يؤدي إلى ضيـاع الأوامـر في حالـة العمـل معـه. وعندمـا يتطلب الأمر تنفيذ الكود البرنامجي بصورة متكررة فإن هـذه الطريقـة تلـبي المطلـوب، وبنفـس الطريقـة يمكـن اسـتدعاء الأوامـر المنفـذة وقبـل العمليـة الحسابية.

٥. إذا كان حجم البرنامج ليس كبيراً كفاية فيمكن أن يضم له عمليـات وهمية (نماذج)، من خلالها تتم عملية نقل التحكم، وأوامر تلـك العمليـات لاتنفذ في الواقع مطلقاً، أي أنه يحوي كمية كبرى من الأوامر الـتي لاتملـك صلة مع منطق عمل البرنامج، وغير ضرورية له. و يجب ألا تكون واضحـة لمن يريد بصورة غير مشروعة مع البرنامج، وكمثال على هـذه الطريقـة فـإن تلك الإضافة يمكن أن تضم أوامر طرق محددة للحماية من طراز int 21h , int 13h أو ذات وظائف متخصصة.. الخ. والمتتبع للبرنامج غير المأذون لـه سيعير الإنتباه لها ويلاحظها ويبدأ بتفحصها وهو بذلك يفقد عنصر الوقـت الثمين.

٦. تغيير بداية البرنامج المحمي كـي لايعـود «Disassembler» قـادراً علـى القيـام بعملـه بصـورة صحيحـة كمـا في مجموعـة الحمايـة مثـل HOTA , COPYLOCK. وهـي إذ تستوعب آليـة حمائيـة في الملـف المحمي فإنها تبدل وتغير بشكل كامل العنوان الأولي للملف EXE.

ونورد فيما يلي مثالاً لذلك الملف EXE قبـل ومـع الحمايـة، و في هـذا المثال يمكن الحديث ليس عن استيعاب الآلية الحمائية بل على العكس عن ضم المسألة المحمية الأولية إلى نموذج آلية الحماية ذاته.

مثال 2.1

النص الأولي للبرنامج S2.exe

```
Sseg segment stack
db 256 dup (?)
sseg ends
dseg segment
eri db  «مرحباً»  , 0
dan dwD
dseg ends
    cseg segment
    assume cs:cseg,ss:sseg.ds:dseg
    z-  proc far
    push ds
    mov ax,0
    puch ax
    mov bx,dseg
    mov ds,bx
    mov ah,09h
    mov dx,offset erl
    int 21h
    ret
    cseg ends
     endz
```

الملف S2.exe نموذج غير محمي.

العنوان

000	4D	5A	20	00	02	00	01	00	20	00	11	00	FF	FF	03	00
010	00	01	83	28	00	00	00	00	1E	00	00	00	01	00	06	00
020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

الكود المنفذ

1F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
200	0E	B8	00	00	50	BB	07	00	8C	DB	B4	09	BA	00	00	CD
210	21	CB	00	00	00	00	00	00	00	00	00	00	00	00	00	00
220	AF	E0	A8	A2	A5	E2	20	24	00	00	00					

الملف S2.EXE النموذج المحمي:

العنوان

000	5D	FD	5A	00	0B	00	00	00	20	00	11	00	FF	FF	04	00
010	AD	14	83	28	20	00	04	00	1E	00	00	00	01	00	06	00
020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

الكود المنفذ

1F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
200	E3	B8	00	00	50	BB	07	00	8D	DB	B4	09	BA	00	00	CD
210	21	CB	00	00	00	00	00	00	00	00	00	00	00	00	00	00
220	AF	E0	A8	A2	A5	E2	20	24	00	00	00	4E	4E	4E	4E	4E
230	06	00	00	00	4E	4E	4E	4E	4E	4E	4E	4E	4E	4E	4E	4E
240	B4	DA	00	FF	FD	FF	FF	F1	16	FF	00	00	00	FC	00	F2

ومن ثم يسير «كود» آلية الحماية بمقياس نحو (5) كيلوبايت .

إن غرس الآلية الحمائية قد أدى إلى تبدلات عناوين البرنامج كما يلي لمجموعة (Copylock).

اسم الحقل	كان	أصبح
«توقيع» الملف	4D 5A	4D 5A
طول الصفحة الأخيرة غير المكتملة	2B 00	F0 00
طول النموذج بمقياس 512 بايت	02 00	0B 00
عدد العناصر في جدول الإنتقال	01 00	00 00
طول العنوان في الفصول	20 00	20 00
النهاية الصغرى للذاكرة في نهاية البرنامج	11 00	11 00
النهاية العظمى للذاكرة في نهاية البرنامج	FF FF	FF FF
تداخل Segment الستيك (SS)	03 00	04 00
قيمة مؤشر الستيك عند تشغيل (SP)	00 01	AD 14
المحصلة التدقيقية	83 28	83 28
قيمة Register1P	00 00	20 00
تداخل Segment الكود (CS)	00 00	04 00
تداخل عنصر الإنتقال الأول في الملف	1E 00	1E 00
رقم أفيرل	00 00	00 00
جدول الإنتقال	01 00	01 00
	06 00	06 00

وكما هو واضح من الجدول فإن Copylock يغير تلك المؤشرات من العنوان مثل عدد عناصره، عنوان النص، عنوان (توجه) أول أمر منفذ. وبالنتيجة فإن المفكك يرتبط بمجال Stack أولى الأوامر المنفذة وذلك لأن عنوان: SS:SP-0004:0003

أما أول أمر منفذ فهو: CSIP:0004:0002

2.1.2 مواجهة طرائق الحماية الديناميكية:

١. يسمح الحساب الدوري للمحصلة التدقيقية، لنموذج أية مسألة موجودة في مجال الذاكرة العملياتية وفي سياق التنفيذ، برصد وتنفيذ أمور محددة هي: ملاحظة المتغيرات في النموذج المحمل وإدخالها. تبين نقاط التدقيق وتكشف محاولات «تجريد» البرنامج.

٢. تدقيق حجم الذاكرة الحرة، والمقارنة مع ذلك الحجم من الذاكرة الذي تشغله العملية عادة أو الذي لقمت به. وهذه الحركة تسمح بتجنب الملاحقة «الخشنة» لبرنامجكم والتي تتم بمساعدة نماذج.

٣. تدقيق محتوى مجالات الذاكرة غير العاملة على البرنامج المحمي والتي تقع ضمن نطاق الذاكرة العملياتية ومن الممكن بنفس الوقت أن يصلها المبرمج، مما يؤدي للحصول على نظام واحد لتشغيل البرنامج.

٤. تدقيق محتوى عوامل التوقف وخاصة 13h و 12h وتحديداً لتبيان مدى وجود تلك القيم المعطاة للمسألة، وقد يكون مفيداً أحياناً مقارنة الأوامر الأولى الناتجة للمنظومة العملياتية المعالجة لهذه التوقفات والإنقطاعات مع الأوامر التي يجب أن تكون عندها. والتدقيق مع تنظيف الذاكرة العملياتية وإعادة تشغيلها يسمح بالتخلص من معظم البرامج المقيمة والموجودة فيها.

٥. إعادة وضع كميات توجيه الإنقطاعات: بعض منها وعلى سبيل المثال 12h و 13h تنسخ في مجال الذاكرة الحرة، وبالتالي يتغير التوجه نحو الإنقطاع وبشكل ملائم، وفي هذه الحالة فإن المتابعة لمؤشرات معروفة تعطي النتائج المرغوبة. فمثلاً تقوم أولى الأوامر المنفذة للبرنامج بنسخ 12h

(٤بايت) في Goh وعوضاً عن الأمر intGoh. وبالنتيجة فإن نص البرنامج لن يعود يحوي أي أمر للعمل مع توقفات وإنقطاعات مع (12 h).

٦. تتبع أوامر السماح والمنع بصورة دائمة فيما يخص الإنقطاعات، وهذا يعيق وضع (منظم) لنقاط المراقبة.

٧. تدقيق ومراقبة زمن تنفيذ أجزاء منفصلة من البرنامج، مما يسمح بإظهار وكشف الإنقطاعات في كتلة النموذج المنفذ.

يمكن تنفيذ كثير من وسائل الحماية تلك بلغة Assembler تحديداً. وتكمن احدى الخصائص الأساسية المميزة لهذه اللغة في أنه لا توجد لها حدود للعمل مع Register والذاكرة وبرنامج الدخل والإخراج.. الخ. ونورد هنا أكثر طرائق الحماية جمالاً للكود البرنامجي بنظرنا، وعلى سبيل المثال طريقة «بابي – ياغي».

المحتوى:

يتحدد ستيك مباشرة في نظام الأوامر المنفذة، مما يؤدي لضياع الأوامر عند تعاملها مع ستيك وكأن البرنامج يجر خلفه كل الآثار المترتبة على ذلك. ومن المهم في هذه الحالة أن تنفذ قمة (JMP) وتهرب من الملاحقة، وذلك حين اقترابها من الكود المستخدم. وتلك طريقة جيدة ومشروطة بعدم تنفيذ العملية المعاكسة أو العودة للوراء.

نقترح فيما يلي مثالاً على طريقة «بابي – ياغي»:

المثال 2.2

من اليسار: قبل تنفيذ الأوامر بإتجاه (CS:000A)، وفي حالة العمل مع (منظم) ويظهر على الشاشة نص البرنامج (مكشوفاً).

من اليمين: في حالة العمل مع (TD) لنفس البرنامج وبعد تنفيذ الأمر بإتجاه (CS:000D).

قبل إعادة وضع Stack	بعد إعادة وضع Stack
CS:0000 1e push ds	CS:0000 1100 adc (bx+sl),ax
CS:0001 b80000 mov ax,0	CS:0002 bc4d02 mov sp,024b
CS:0004 50 push ax	CS:0005 f30000 rep add(bx+sl),al
CS:0005 0e push cs	CS:0008 0000 add (bx+sl),al
CS:0006 17 pop ss	CS:000A 8be3 mov sp,bx
CS:0007 bb0a00 mov bx,0a	CS:000C 50 push ax
CS:000A 8be3 mov sp,bx	CS:000D 52 push dx
CS:000C 50 push ax	CS:000e bbbe4b mov bx,4bbe
CS:000D 52 push dx	CS:0011 8edb mov ds,bx
CS:000E bbbe4b mov bx,4bbe	CS:0013 b409 mov ah,9h
CS:0011 8edb mov ds,bx	CS:0015 ba0000 mov dx,0
CS:0013 b409 mov ah,9h	CS:0018 cd21 int 21h
CS:0015 ba0000 mov dx,0	
CS:0018 cd21 int 21h	

يجب عدم نسيان Stack قبل الخروج من البرنامج، ويجب أيضاً أخذ الخصائص التالية بعين الإعتبار: في حال (20h) يتم الإنزياح دائماً نحو الأعلى أثناء تنفيذ أمر (push) لذلك من الضروري تخصيص لاأقل من (20h).

والمثال التالي يوضح كيف يمكن تنفيذ ما يمكن تسميته «الركض أمام القاطرة» أي استخدام ذات الطريقة لأجل «الكود» المستخدم قبل العملية الحسابية.

المثال 2.3

```
mov dann,sp        المحافظة على الستيك
mov sp,48h
```

```
mov cx,4
wosw: push cs:(bx)        تعبئة الستيك
sub bx,2                  يتشكل في النص كود مفكك
loop wosw                 الإنتقال للكود المشكل من جديد
jmp p1

db 20 dup (90h)
p1: nop
db 10 dup (90h)
nop
mov bx.sseg               إرجاع الستيك
mov ss,bx
mov sp,dann
ret
cseg ends
end z
```

المثال 2.4

نورد هنا نص البرنامج الذي في سياق التنفيذ إلى مجال معطيات الأوامر المنفذة والمسؤولة عن عرض المعلومات على شاشة Display، وذلك بعد تنفيذ الأوامر التالية :

```
mov ah,09h
int 21h
ret
```

```
sseg segment stack
db 256 dup (?)
sseg ends
dseg segment
er3 db  « Pasynkova Natalia$» ,0
er4 db          « priwet$» ,0
er5 db          62h,0A8h,5Ah,0ADh,5Ch,0
dseg ends

cseg segment
assume cs:cseg,ss:sseg.ds:dseg
z       proc far
        push ds
        mov ax,0
        push ax
        mov bx,dseg
        mov ds,bx
        mov cx,5
mov si,0
a: mov al,er5 [si]
add er3[si].al
inc sl
loop a
```

تجدد الأوامر

جمع سطرين

والنتيجة تكون

```
: mov ah, 09h
: int 21h
```

: ret

```
mov bx, offset er3
mov dx, offset er4
```

عنوان السطر الموضح على الشاشة :

```
push ds
```

تحضير Stack للإنتقال :

```
push bx
ret
```

الإنتقال في مجال المعطيات .ha er3 :

```
cseg ends
end z
```

وعندما يدقق مالك الملف EXE فإنه يرى التالي:

200	1E	B8	00	00	50	BB	03	00	8E	DB	B9	05	00	BF	00	00
210	8A	84	1B	00	00	84	00	00	46	E2	F5	BB	00	00	BA	13
220	00	1E	53	CB	B4	09	CD	21	CD	00	00	00	00	00	00	00
230	52	61	73	74	6E	72	67	75	65	77	20	53	65	72	67	65
240	69	24	00	70	72	69	77	69	74	24	00	62	A8	5A	AD	5C

ويظهر هذا البرنامج وكأنه مدخل يعطي على الشاشة كلمة (PRIWET) وينهي عمله. وإذا نظرنا إلى نصه كأنه لايمكن رؤية الأوامر المعدة للعمل مع الشاشة. وفي هذه الحالة فإن محاولة تنقيح العلامة الأصلية للمؤلف ستؤدي لتغيرات لا يمكن توقعها في حزمة الأوامر المنفذة.

2.2 قسم خصائص الوسط :

وهذه الوحدة مسؤولة عن تعارف البرنامج المحمي مع الوسط المحيط، ويكون ذلك أما في البرنامج المحمي ذاته أو وحدة (قسم) خاصة. المختص

يدخل في توليفة البرنامج – الملف المحمي – وفي خصائص الوسط التي تم تحديدها وتميزها، أو بالعكس يلحق الوسط المحيط كما هو حال (DOS) بما تحت البرنامج المحمي إذا لم يكن الأخير قادراً على القيام بوظائفه في محيط طبيعي.

ماهي مواصفات وسمات الوسط؟ وماهي الأمور التي يجب على البرنامج أن يتأقلم ويعتاد عليها؟

قد تكون تلك المواصفات الخطوط التالية:

- المحيط الجهازي والمعدات للأجهزة الحاسبة.
- المحيط البرنامجي.
- المواصفات السيكولوجية – الفيزيائية لمشغلي التجهيزات.

وإذا كان تفرد الوسط غير كافي فيجب إيصاله لحد الكفاية.

2.2.1 خصائص المحيط الجهازي – التجهيزات والمحيط المرتبط بها –.

سنتحدث هنا عن مواصفات أجهزة الحاسوب التي لايستطيع المستخدم تغييرها بطريقة برنامجية ومنها على سبيل المثال خصائص BIOS والمعدات الخاصة الإضافية وهندسة بناء (منظور وتكامل) الحاسوب أو مايسمى معمارية الحاسوب.

١. **معمارية الحاسوب وتكامله:** من حيث وجود معدات A, B, C, D والتي تسم نمطية الجهاز، الفيديو ادابتر (EGA, CGA, VGA) وحجم الذاكرة العملياتية...الخ.

يمكن للبرنامج أن يحصل على إمكانية الوصول للمعدات بالطرائق التالية:

يمكن مخاطبة أي من لوحات الإدخال / الإخراج والمناسبة والملائمة للمعدات الملحقة، أو لأي عنوان في الذاكرة العملياتية. في المرجع [17] أوردنا وبالتفصيل كل إمكانيات الوصول الموجودة، ونورد هنا العناوين الرئيسية مع وضع الموجز لكل منها:

ميكرو مخطط intel8255 ووظائف هذه الوحدة هي:

- تعلم عن وحدة القواطع في جسم المنظومة.
- تأخذ الإدخال مع كلاميناتور الكمبيوتر (الحاسوب).
- تقود عدد من التجهيزات والمعدات.
- ريجستر (رقم) ميكرو مخطط intel8255.
- البوابة: الميناء A (GOH) – قراءة فقط.

عندما تكون الخانة السابعة تساوي الصفر في الميناء فالخانة من (٠ ÷ ٧) للـ PC, AT, PCV, XT هي ثمانية للكود مع كلاميناتور وعندما تساوي الخانة السابعة في الميناء (١) فهي بالنسبة لـ (PC) في البوابة A تكون:

الخانة	القيمة والمعنى
0	لا يوجد مخزن في الديسكيت
1	لا يُستخدم
2-3	عدد مصارف الذاكرة في المنظومية
4-5	نوع Display (11 مونو كروم , 10 ملون 80*25 , 01 ملون 40*25)
6-7	عدد المخزن في الديسكيت

البوابة – البورت (G1h) B قراءة وكتابة:

الخانة	الحاسوب	المعنى
0	pc, xt, pcjr	يتحكم بالقنال 2 و 8253
1	pc, xt, pcjr	الخروج للديناميك
2	pc	اختيار البوابة المحتواة (C)

	pcjr	النظام الرمزي الأول (1)
		بيانى (D)
3	pc, pcjr	محرك الشريط متوقف
	xt	اختيار محتوى البوابة (C)
4	pc, xt	0 - السماح
		1 - منع الديناميكا ومحرك الشريط
5	pc, xt	0 - السماح بالخطاء
		ثقوب التوسع
6	pc, xt	السماح بساعات كلافياتور
5-6	pcjr	اختيار الديناميكا
		الشريط 8253,01 ÷ 00
		10- دخل / خرج
		11- ميكرو مخطط 76496
7	pc	اختيار محتوى البوابة A
	pc, xt	تأكيد كلامنياتور

البوابة C(62h) قراءة فقط.

إذا كانت الخانة الثانية تساوي (0) بالنسبة (PC) في البوابة (B) أو الخانة الثالثة تساوي (0) بالنسبة لـ (XT) فإن:

الخانة	الحاسوب	المعنى
0	pcjr	فقدان أول رمز مدخل
1	xt	يوجد معالج رياضى خاص
	pcjr	توجد خارطته
2	pcjr	توجد خارطته

2-3	xt	عدد مصارف الذاكرة في المنظومة
3	pcjr	0 - 128k ذاكرة
4	pc, pcjr	إدخال الشريط
	xt	لاتستخدم
5	pc, xt, pcjr	خروج القنال 28253
6	xt, pc	1- تدقيق الأخطاء لثقوب التوسع
	pcjr	1- المعطيات مع كلافياتور
	pc, xt	1- المراجعة الأولى لأخطاء الحساب
	pcjr	0- كابل الكلافياتور موصول

البوابة C(62h) قراءة فقط ؟

إذا كانت قيمة الخانة الثانية تساوي (0) بالنسبة للـ (PC) في البـورت (B) أو الخانة الثالثة تساوي (0)بالنسبة لـ (XT) فإن :

الخانة	الحاسوب	المعنى
0-1	XT	نوع الديسبلاي: 11- مونوكورم، ١٥ ملون 80*25 , 01 ملون 40*25
2-3	XT	عدد 1÷ 00 الخ..
4-7	PC,XT	نفس الأمر للخانات المحددة

علماً أن الصفر (0) في حالة Register يطابق وحدة الفصل (off)، تحفظ الآلات الحاسبة الالكترونية المتوضعة مشاركة مع AT.

وفقاً للمرجع [17] المعلومات الخاصة بالمخطط الميكـري MC 146818 والعائد لشركة (MOTOPOLA) بما ينسجم ويتلائم مـع حاسـبات الزمـن الحقيقي. يوجد في «الميكرو مخطـط» 64 ريجسـتر مرقمـة مـن (00) وحتـى (3FH). من أجل قراءة Register يجب في البداية إرسال رقمه في البـورت مع العنوان (70h) ومن ثم قراءته من خلال (71h).

رقم التسجيل	الإستخدام
0h	نوع
2h	نوع – مخزن تحديد الديسك
4h	بيريفيري
5h	الخانة الصغرى
6h	الخانة الكبرى
7h	الذاركة العامة – الصغرى
8h	الذاكرة العامة – الكبرى
9h	ذاكرة أكثر من ١ميغابايت – الذاكرة الأصغر
1h	ذاكرة أكثر من ١ميغابايت – الذاكرة الكبرى

توجد في النهاية الثانية لبايت الذاكـرة معلومـات حـول النمـوذج IBM PC ووفقاً للعنوان التالي:

FFFFE في ROM BIOS

الحاسوب	الكود
PC	FF
XT	FE
PCJV	FD
AT	FC

إلى ذلك فإنه يمكن الحصول على قائمة المعدات الموجودة في الآلات الحاسبة الإلكترونية بواسطة البرنامج أما من Register (AX) بعد تنفيذ (11H) أو مباشرة بالعنوان 0:0410.

عملياً فإن قائمة المعدات موضحة في دليل الكترونيات شركة IBM PC ولسهولة إستخدام ذلك الدليل في التطبيق العملي نورد المحتويات الرئيسية لما أشرنا إليه:

الخانة	المعنى الوظيفي
0	لاتوجد تجهيزات الديسك
1	8087 براسيسور النسخ
2-3	حجم الذاكرة العملياتية
4-5	النظام الفاعل والأولى للديسيمبلر
6-7	إدخال الديسك فقط
8	DMA موجودة
9-11	عدد (بورت) RS-232
12	يوجد (ادابتر) العاب
13	برينتر متواصل
14-15	كمية برينتر (عدد)

وفي حال استخدام هذه المعلومات بهدف الحماية، والضرورية إما لواضعها أو من حزمة الأجهزة ذاتها، يجب تذكر الكلمات المحتواة حسب العنوان 0:410 في جسم الملف المحمل عند التشغيل الأول ومن ثم إجراء المقارنة بعد التشغيل. عندما لاتعود قائمة المعدات مطابقة لبدايتها الأولى فيجب إعتبار أن البرنامج تمت سرقته. ولهذا المدخل نواقص عديدة، الأساسية منها:

• يعتبر البرنامج جميع الآلات الحاسبة الألكترونية التي لديها مثل هذه القائمة للمعدات وكأنها ذاتها، أي «من الأهل».

• إذا جرى تغيير في الآلة الحاسبة وفي البرنامج، فإنه يعني نهاية البرنامج وموته إذا كان ذلك ملحوظاً مسبقاً، وبغض النظر عن أنه لم تجرِ عملية نسخ غير مشروعة.

٢. مواصفات وسمات BIOS، المحصلة التدقيقية (الإختيارية) وتاريخ الصنع:

في ذاكرة الحاسبات الألكترونية يوجد RAM BIOS، بالنسبة للمحصلة التدقيقية بداية من العنوان F000:E058 وحتى العنوان F000:FFFF. وفي العنوان F000:FFF5 يحدد عادة تاريخ صنع BIOS. كما هو معروف يوجد مغزى واحد في عناوين الحاسبات الألكترونية الحديثة من طراز (XT)، لذلك من المفيد إجراء تدقيق وتفتيش بنفس الوقت على المحصلات التدقيقية لجميع ماهو موجود منها بما فيها BIOS، فيديوتير... الخ. مع أن هذه الطريقة لاتسمح كفاية بحماية ضمانات البرنامج في إطار السلسلة.

٣. السمات الديناميكية لأجزاء الحاسوب وتناسبها:

غالباً ما تكون الأجزاء الألكتروميكانيكية المتماثلة والعائدة لحواسب مختلفة من سلسلة إصدار واحدة، ولدينا عدد من السمات المؤقتة والزمنية، يمكن قياس ذلك الزمن وإعتباره خاصية لمحيط التجهيزات وإستخدامه على أساس ذلك. تكمن الصعوبة الأساسية في استخدام السمات الديناميكية بإرتباطها بدرجة الحرارة وزمن العمل.. الخ، مما يجعل من الصعوبة بمكان الحصول على سمات مستقرة كفاية ومتفردة، لذلك فإن هذا النموذج يحمل طابعاً نظرياً صرفاً.

٤. التبديل الخاص في BIOS:

أبسط طريقة لتبديل وتغيير BIOS انما تقوم في تغيير صندوق واحد أو أثنين، ولهذا الغرض يكفي القيام بعملية تفكيك BIOS، في (SR) Sourser يجب انتقاء الخانة المناسبة من القائمة الناتجة والتي محتواها لايترك بصماته على عمل الحاسوب. عملياً في نهاية ROM BIOS توجد مقاطع غير مستخدمة، ولذلك مرة أخرى يتم المبادل الملائم وهكذا يتحول

حاسوبكم لشيء فريد بذاته. والمعلومات حول ذلك يجب أن تدخل في برنامج الحماية وهذا أمر سهل ويكفي أن يُلحظ في البرنامج كود مطابق وقراءة العناوين المختارة ومن ثم إجراء المقارنة.

مثلاً إعادة برمجة الخانة في العنوان F000:FF4C على قيمة غير الصفر ذلك لأن البداية الأولى هي صفر. يوجد نموذج آخر أكثر رهافة وفقاً للتبديل في BIOS وعلى سبيل المثال يمكن تغيير معالجة الإنقطاع 13h بإضافة تابع جديد للمعالجة ومن ثم استخدامه في البرنامج المحمي. نضع أمراً بإعتباره يمثل أولى الأوامر المسؤولة عن معالجة 13h فإذا كانت قيمة الريجيستر AH - والذي من خلاله يتحدد رقم التابع - أكبر من القيمة الأعظمية فإن تحديده مرة أخرى يمكن من خلال الرقم الإنتقائي للتابع. ومن الضروري استخدام هذا الرقم الإنتقائي في البرامج المحمية من أجل تحديد القيمة الأكبر من الأعظمية من خلال المبدل.

وفي هذه الحالة فالآلة الحاسبة الألكترونية ذات BIOS مبدل يمكن أن تعمل مع جميع الحسابات المعيارية نظراً لغياب التوجه فيها نحو 13h مع أرقام التوابع الأعلى الأعظمية. وبرنامجكم سيعمل على هذا الحاسوب تحديداً لأنه لايوجد غير قادر وبصورة صحيحة معالجة دخل/خرج برنامجكم. وما ذكرناه طريقة جيدة للحماية إذا قمتم بأنفسكم بوضع مجموعتكم في حاسوب صاحب الطلب، والقضية تكمن في أنه يتوجب عليكم فعل ذلك أمامه وكشف جسم جهاز الحاسوب أمام عينيه وتغيير في ميكرو الحاسوب. والقيام بهذه العملية معقد للغاية وهو الأمر الذي يخفف من القدرة التنافسية لهذه الطريقة الحمائية للبرامج، ويعتبر ذلك من العيوب الأساسية.

٥. المعدات الخاصة :

يتم غالباً استخدام معدات خاصة لتقوية الحماية ميكرو سخيم منطقية، إضافة مربوطة مع الحاسوب على التوازي أو التسلسل، إينترفيس واضح لجميع «الكود». وفي هذه الحالة ينهض سؤال حول حماية التجهيزات الخاصة ذاتها من النسخ والتقليد وتعتبر أكثر الطرق بساطة في هذا الإطار

تغطية تلك المعدات بطبقة من القطران الهيبوكسيدي، ولكن تلك الطريقة لاتحل المشكلة تماماً فقد يوجد هواة بإمكانهم قص تلك الطبقة وإزالتها تدريجياً، أكثر المعدات شهرة في هذا المجال هو الأنترفيس الجهازي مع مجموعة الإستخدام لضمان منع النسخ والتقليد غير المشروع.

مقاييس وحدة الأنترفيس (8 ,3 ,1 cm) وهي مخصصة لأجل الوصل على التوازي مع البوابة، وقد اتبع هذا الأسلوب أيضاً مصممو حماية مجموعة Autocad Co وهذا الربط يؤمن نقل «واضح وشفاف» لجميع المعطيات على التسلسل. وكل انترفيس جهازي أو مجموعة منه تستجيب لخانة مفتاحها التسلسلي، على المستخدم أن يعد مجموعة برامجه للعمل مع الأنترفيس الجهازي. وهكذا فإن مجموعة الأجهزة ستقوم بتأدية وظائفها بصورة طبيعية بوجود انترفيس الجهازي المطابق لها.

- وجود الربط يسهل عملية إيصال المجموعة لصاحب الطلب وخاصة إذا تم ذلك عن طريق البريد.

- محاولة فتح الإنترفيس الجهازي ستؤدي إلى تحطيم المعدات.

لم تجد هذه الطريقة استخداماً جماعياً لغاية الوقت الراهن لسببين أساسيين:

* تعقيد عملية معالجة الميكروسكيم الوطنية دون فتح جسم الحاسوب.

* ضرورة تصميم وحدة حماية نوعية من «المجرد» ، وهذا هام جداً لأن منهم منطق العمل مرتبط بفهم البرنامج، وذلك الجزء منه المسؤول عن الانترفيس.

٦. وجود بعض العيوب في الحاسوب أو في MD:

بعضهم يحفرون ثقباً فيه وآخرون يستخدمون الليزر لمتابعة مرادهم ومن ثم يقومون بحفظ أرقام المقاطع ذات الخلل في البرنامج، ويتابعون محاولة إيجاد نص طبيعي في القطاع المعاب للبرنامج المحمي وذلك من خلال تدقيق «كود» الإنهاء، والناقل بدون شوائب (نسخة غير مشروعة) يسمح

بإيجاد نص طبيعي في مكان الخلل، أما الناقل مع شائبة فلا يسمح بذلك، وعلى هذه الطريقة سار مصممو الحماية KD 4 وأكثر طريقة منتشرة لنزع مثل هذه الحماية هي في البرنامجية لخصائص الديسك.

يمكن تلافي النقص إذا تمت مخاطبة المعدات الخارجية معنونة وموجهة ومباشرة وليس من خلال DOS. إجراءات الحماية تلك يمكن إنجازها فقط لأنواع الديسك الذي يمكن نزعه، وهي تسمح بالوصول لسطحها المغناطيسي، مما يعطي مجالاً لإكساب ذلك الديسك خصائص متفردة عن طريق وضع علامات فيزيائية محددة على الناقل، يمكن أيضاً ايجاد (نقل) نسخة دقيقة جداً للديسك مع كافة خصائص الأصل، لا تسمح الأقراص غير القابلة للفك بإمكانية الوصول لسطحها المغناطيسي من قبل المستخدم.

تجري العمليات في الناقل من خلال المعدات المربوطة مع الديسك في مجموعة واحدة، ولإجراء نسخة مطابقة من الضروري التعامل أولاً مع نظام للحماية. ومع إنخفاض الأسعار من الممكن نشر البرامج القيمة على نواقل متنوعة ومن ثم تأمين تركيبها في الحاسوب.

2.2.2 خصائص المحيط البرنامجي:

سنتحدث هنا حول الخصائص التي تتشكل من خلال الطرائق البرامجية، ومنها مكان توضع الملف في الديسك، التقييمات الخاصة في الملفات، تبدلات DOS, النموذج الخاص للديسك..الخ

أولاً – موضع التوضع الفيزيائي للملف المحمي:

يدخل حقل POSWRK FCB من أجل الإحتياط، انظر المرجع (٣٥). عملياً بعد فتح الملف فإن المنظومة تحوي هنا في هذا الحقل عنوان التوضع الفيزيائي لقطاعات الملف. احتمالات نسخ الملف من ديسك لآخر مع الإحتفاظ بالعنوان المنوه عنه قليلة جداً، الحديث يجري هنا طبعاً حول الفينتشستر وإذا أمكن النسخ فيجب نسخ كامل الديسك، والقرص الذي يتم النقل إليه يجب أن يكون مماثلاً للأصل.

يحتمل أن يكون تتابع وحدة الحماية كما يلي:

يضاف في بداية البرنامج أدابتر لإستدعاء ملحق البرنامج المسؤول عن الحماية. يمكن القيام بإجراء بعض التعديلات في البرنامج من أجل الموثوقية وكذلك بعض الإستدعاءات. المؤشر الوحيد لملحق البرنامج هو الملف المحمي ذاته ويجب أن يحوي الحقول التالية (انظر المثال (2.6)):

- حقل الرمز (psch) المقياس (N) بايت (خانة)، يحدد قيمتها المستخدم، الحقل المخزن ذو توليفة رموز عددها (N).
- حقل الإطلاق الأول (pz) المقياس: الكلمة. المخزون الأولي (0).
- حقل وصف الملف (FCB) هذه التسمية مأخوذة من المرجع (15).

أما اللوغاريتم عمل ملحق البرنامج فهو كما يلي:

١. تتوضع الحقول ENAME, FNAME الحاوية على اسم الملف في وحدة وصف الملف FCB. تنتقل قيمة ومغزى هذا المؤشر من البرنامج المحمي إلى الملحق.

٢. فتح الملف.

٣. تدقيق وإختبار محتوى PZ إذا كان يساوي (0) فننتقل للخطوة (7).

٤. مقارنة محتوى PSRED وDOSWRK من FCB إذا لم تتطابق ننتقل للخطوة (6).

٥. إغلاق الملف والخروج من الملحق.

٦. تجسيد وإنهاء العمليات الجوابية. إنهاء المعلومات في الناقل المغناطيسي أو إنهاء (تحطيم) ذلك البرنامج الموجود على الناقل والإنتقال للخطوة (5).

٧. قراءة الملف من الناقل المغناطيسي والبحث فيه عن توليفة (N) رمز والمعطاة في PSCH. إذا لم نجدها ننتقل للخطوة (6).

٨. توجد الحقول PZ وPSRED بعد PSCH نضع PZ في (1). ننقل محتوى DOSWRK من FCB إلى PSRED.

٩. نسجل في الملف المعطيات المدققة، والإنتقال للخطوة (5).

نورد فيما يلـي مثـالاً لتنظيـم الحمايـة بلغـة Assembler مـع كـود ذو تأطير ذاتي عام:

المثال 2.5:

```
dseg:segment
uuu db              علامة موازية : «$ عودة التحكم»
even                         للبحث عن mestox
mestox dw 0    مكان التوضع الفيزيائى للملف في لحظة وضع المجموعة
imj2 db 0                    : FCB
        db» zap«             اسم الملف :
        db» exe«                    التوسع :
        db 15 dup (?)   مكان التوضع الفيزيائى للملف :
mestoy dw (?)
        db 9 dup (?)
        db 20 dup (?)
dseg ends
cseg segment
   assume cs:cseg,ds:dseg,es:dseg
   z     proc far
         push ds
         mov ax,0
         push ax
         mov bx,dseg
         mov ds,bx
         mov ah,0fh
```

```
mov dx,offset imj2
int 21h            : فتح الملف
mov ax,mestox: : مكان التوضع الفيزيائى للملف
sub ax,mestoy:
push ds            : وحدة العنوان لأجل
push cs            : الأوامر
pop ds
mov bx,offset oper
                   : الأوامر
mov ds:[bx],ax
mov al,0CDh    :INT
add ds:[bx],al]
add bx,1           : العنوان التالى
mov al,21h
add ds:[bx],al         : 21h
المجموع                : int 21h
```

يكون التنفيذ صحيحاً إذا كان توضع الملف على (MD) أيضاً كما هو حال المجموعة.

وفي الحالة المعاكسة يتشكل أمر غير متوقع.

```
pop ds
    mov dx,offset imj2
    mov ah,10h
    int 21h            : الملف مغلق
```

```
push cs
pop ds
mov dx,offset prod
mov ah, 25h
mov al,18h
oper dw0
ret
```

```
prod: push ds    : (18h) برنامج معالجة
push bx          : تصل العملية لهنا فقط عندما
push ax          : بصورة         تتم عملية
push bd          : صحيحة للأمر
push es                 (int 21h)
push cx


aa:  pop cx
     pop es
     pop dx
     pop bp      : (18h) وحدة معالجة الإنقطاع
     pop ax
     pop bx
     pop ds
fin:             ret
     cseg ends
end z
```

ثانياً – نظام التوضـع الفيزيـائي لملفـات المجموعـة المحميـة في الديسك المغناطيسي:

كل شيء مماثل للنقطة السابقة، والفارق يكمن في أنه لايجري التدقيـق في ملف واحد بل في عدة ملفات مباشرة بما فيها ملفات المعطيات، ويؤخـذ بعين الإعتبار أيضاً توضع الملفات بالنسبة لبعضها البعض، وبالمناسبة فـإن تدقيق مكان توضع ملفات المعطيات يمنح العمل أكثر ضمانـاً وثقـة أكـبر، ذلك لأن التعامل معها عادة يجنب مـن ملاحقـة ومتابعـة فقـرات النمـاذج المضادة للفيروسات.

ثالثاً – مدى وجود جداول، ملفات، Driver سـهلة التحقيـق في الحاسوب:

يعتبر تواجد الجداول والملفات عاملاً هاماً في تـأمين طريقـة سـهلة يتـم من خلالها نزع الحماية لدى الناس نظرات مختلفة لعاداتهم. أحدهم مثـلا قـادر خـلال يـوم واحـد أن يـترك XTREE المتـأقلم معـه وأن ينتقـل إلى (NC). وآخر وفي حال ظهور XTGOLD, XTPRO يفكر طويـلاً قبـل أن يغير المجموعة. ويمكـن طبعـاً لآخـر أن يخـاطر وأن يعتمـد في ذلـك علـى إحسـاس مسـبق وهـذا عـائد للنفـس البشـرية. وعمليـاً فـإن زمـن حيـاة الاحتمالات المختلفة لأي مجموعة لا يتجاوز العام أو العامين، والمجموعـة المحمية ستلقى الموت الطبيعي، وقبل أن يحـدث ذلـك التغيـير، سـيكون البرنامج مضطراً عندها للإنتحار. ولكن إذا لجأتم لطرق الحماية فتذكروا:

كي تنظروا في قائمة البرامج المضطـر لهـا المـالك المحتمـل لمجموعتكـم مستقبلاً فليس ضرورياً تماماً العودة إلى فينتشيستر فـأكثر المعلومـات أهميـة لهذا الكمبيوتر والطرق لهذه البرنـامج (AUTOEXEC.BAT) موجـودة في الذاكرة العملياتية. ويمكن لنظام DOS ومن أجل إحتمـال محـدد أن يقـوم بذاته مبدئياً بإيجاد تلك العناوين من الذاكرة العملياتية مسـتخدماً في ذلـك أي من المنظمات أوبرنامجاً خاصاً.

رابعاً – وجود سلسلة استدعاء معطاة في البرامج المحددة قبل إطـلاق المجموعة المحمية:

الأحكام المماثلة للنقطة السابقة.

خامساً – برامج المنظومة العملياتية التي طرأ عليها تبديل أو تغيير خاص:

يعتبر هذا النموذج الأكثر شهرة بين المصممين غير المحترفين، عادة من أجل إجراء تبديل يتم استخدام ملفات المنظومة العملياتية، ويقوم المصمم بتغيير خانة واحدة أو اثنتين في تلك الملفات ومن ثم يقوم بالاختبار. وكل برنامج يستطيع تغيير خواص الملف هو قادر على إجراء عملية تفكيك للملفات وإيجاد الخانة المناسبة كي يتم لحظها، عادة تعتبر ملفات المنظومة العملياتية محمية من التسجيل والكتابة ومغلقة ومن أجل ملاحظة أحدها يكفي:

- تغيير خواص الملف وجعله عادياً.
- كتابة قيم مطابقة في مكان محدد مسبقاً.
- إعادة الخواص القديمة للملف .

الخانة المبدلة تجعل الوسط البرنامجي فريداً، وعلى البرنامج المحمي بهذه الطريقة أن يدقق في تواجد خانته الخاصة به في ملفات DOS أو في الذاكرة العملياتية، هذا إذا كنتم قد حملتم المنظومة العملياتية بخططكم. والأكثر ملاءمة هو تبديل DOS من حيث المبدأ وكما ورد في القسم 2.2.1 جرى الحديث حول طرائق تبديل وتغيير BIOS والفارق يكمن في ضرورة معالجة (12h).

لوغاريتم عمل البرنامج:

١. يمر البرنامج المحمي من خلال المبدل الذي يبدل الأوامر للبرنامج والسابقة للأمر INT 21h وذلك في مقطع نقل القيم إلى AH Register أو أي واحد آخر، مثلاً عوضاً عن (9) يبدل دائماً وفي كل مكان بـ (100) و(10) بـ (101).. الخ.

٢. يقوم انستالياتر المجموعة المبدّلة في الحاسوب بالعمليات التالية: يغير خواص ملف المنظومة العملياتية، وبذلك يسمح لنفسه بكتابة ما يريده. وعوضاً عن معالجة (21h) يقوم بالإنتقال غير المشروط (jmp) إلى مجال المعطيات حيث يعيد استرجاع الأولى بدلاً من أمر الانتقال لأمر آخر

ينص على مقارنة (100) مع محتوى AH Register. إذا كـانت رقـم AH أكـثر مـن (100) فيقـوم بحـذف قيمـة (100) مـن AH Register. ينجـز توضع الأوامـر المسـؤولة عـن الإنتقـال غـير المشروط للأمـر الثـاني لمعالجـة (21h). يعيد خواص الأولى IBM BIO.COM.

٣. ينسـخ انسـالياتر المجموعـة في حاسـوب طـالب DOS والخـاضع للتبديل المذكور في الخطوة (١)، كمـا هـو واضـح مـن العـرض السـابق فـإن التبدلات المشابهة لـ DOS لاتنعكس على عمله مع تراتيب أمان برنامجيـة أخرى، ورغم كل الاحتراس فمن السهولة بمكان نقل المنتوج المحمي بهـذه الطريقة من آلة حاسبة الكترونية إلى أخرى. ويكفي نسخ DOS مـع المنتـج والقضية منتهية. وأمر واحد Copy يؤدي بكل الجهود المبذولة.

٤. تدقيق وجود معطيات فريدة، مثلاً تاريخ وضع الديسك.

٥. الطرائق غير المعيارية لتنظيم المعطيات على النواقل المغناطيسية.

تلعب دراسة الدقائق في عمل المعدات المخزنة والـ Driver الخاص بهـا دوراً كبيراً في تنظيم الطرائـق اللامعياريـة لتنظيـم المعطيـات علـى الحوامـل المغناطيسية. البرامج المعيارية عادة تنفذ نسخ الحوامل المغناطيسـية بصـورة سليمة، وإسـتخدام اللامعياريـة -ولكـن المكنـة- مـن الطرائـق لتنظيـم الحوامل المغناطيسية، وتُمكّن من حـذف واستبعاد نسـخ الحوامـل بـالطرق المعيارية. ولأجل القيام بالنسخ يجب إيجاد برنامج منفصل.

غير أن الجانب السلبي لهذه الطريقة تكمن في الإمكانيـات للتصـورات اللامعياريـة للمعلومـات علـى النواقـل المغناطيسـية وبالنتيجـة الحمايـة الضعيفة أيضاً. يستخدم غالباً من أجـل الديسك المغناطيسـي عمليـة تغيـير ترقيم القطاعـات، تغيـيرطول وعـَدد القطاعـات علـى السـكة، تغيـير عـدد السكك.

يمكن في المرجع (16) إيجاد مثالاً على البرنامج اللامعياري.

للديسكيت وجه واحد أو أثنين، وكل جهـة مقسـمة إلى عـدد (N) مـن السكك (عادة نحو ٤٠) كل سكة مقسمة إلى قطاعات. تتم عمليـة Format في أحد السكك وذلك بمقـدار مـايكون الفـاصل مـا بـين السـكة والعلامـات

الموجهة مترابطاً، يجب تقديم المؤشرات التالية من أجل عملية التأخير العام (Format السكة):

- رقم السكة.
- الجهة.
- رقم القطاع.
- كود طول القطاع (0) – يطابق 128 بايت, 1 - 256, 512 - 2 , 3 1024 - بايت.

ولجعل الديسكيت فريدة يمكن في مرحلة التأطير العام تنويع تلك المؤشرات مثل رقم القطاع (كما في المثال السابق) وكود الطول (يساوي هذا في المثال 512-2 بايت).

في الحالة الأولى: إذا أشير خلال عملية التأطير Format إلى رقم غير موجود للقطاع، أي رقم أعلى من الأعظمي المسموح به فيمكن الحصول على قطاع فريد لايمكن قراءته أوتسجيله. وعند التعامل مع قطاع كهذا – إضافة لوظيفة القراءة أو التسجيل – نستنتج كود الطوارئ والنهاية العاجلة الطارئة.

وكمثال على البرنامج المؤطر لسكة واحدة نورد المثال 2.6:

المثال 2.6:

تأطير صحيح للقطاعات 1 و 7 ذات (38) سكة title.

```
stack segment stack
db 64 dup (?)
stack ends
code segment
assume cs:code,es:code
```

```
sektor db 0,0,10,2 : 1sektor
```

يشار لرقم غير موجود بالنسبة للقطاع (الرقم 10) :

```
      db 0,0,2,2 : 2
      db 0,0,3,2 : 3
      db 0,0,4,2  : 4
      db 0,0,5,2  : 5
      db 0,0,6,2  : 6
      db 0,0,11,2: 7
```

النسبة للقطاع يشار لرقم غير موجود هو الرقم (11) :

```
      db 0,0,8,2  : 8
      db 0,0,9,2  : 9
format proc far
      push ds
      sub ax,ax
      push ax
      lea bx.sektor
      push cs
      pop es
      mov ch,38
```

اختيار السكة :

```
      mov cl,1
      mov dx,0
      lea di,sektor
      mov al,9
powtor:
```

```
        mov es:[di],ch
        adddl,4
        dec al
        jnz powtor
        mov ax,50 1h
        int 13h
        ret
    format endp
    code ends
        end format
```

يمكن استخدام «كود» الإنهاء الحاصل كمفتاح في حال الحماية، مما يسمح بتوجيه البرنامج نحو المنافس.

المتغير الثاني في عملية format السكة هو كود الطول، يتحدد في مجرى هذه العملية مقياس كل قطاع إفرادياً وبصورة منفصلة، مما يعطي الإمكانية لنشوء عدد من القطاعات يجب تغيير مؤشر قياس القطاع في جدول قاعدة الديسك خاصة عند القيام بعملية التسجيل / القراءة. قاعدة الديسك [16] جدول مؤلف من (11) خانة يحوي عدداً من المؤشرات الضرورية للعمل مع مدخلات الديسك، بما فيها معلومات محددة حول format الديسك ذاته. عنوان الجدول موجود في مساحة التوجيه حسب التوجه (120)، تداخل الخانة المسؤولة عن طول القطاع نسبة لبداية الجدول يساوي (3). لا يمكن بالوسائل العادية لـ DOS قراءة هذه القطاعات مما يمكن من إخفاء المعلومات الضرورية فيها.

نورد أدناه مثالاً للبرنامج «توربو – سي» الذي يشكل (38) سكة علماً أن القطاع الثاني للسكة يمتلك مقياساً لامعيارياً. أما الثالث فيحوي رقماً غير موجود، وعلى ذلك ينجز البرنامج عملية «تسجيل/قراءة» في القطاعات وطباعة كود إنهاء العملية.

المثال 2.7:

```
#include .h
        main 0
        {
        union REGS rl:
        struct SREGS rg:
        unsigned intj:
        int n:
        char a[1024]:            /* بوفر للقراءة */
        char a[1024]=     تدقيق التسجيل والقراءة في السكة
        int nsek
        int smes:
rl.h.dl=0;              /* الجهاز a */
rl.h.dh=1;              /* رقم الترويسة */
rl.h.ch=38;             /*رقم السكة */
rl.h.cl=j;
rl.h.ah=3;              /* تابع التسجيل */
rl.h.al=1;
                        /* عنوان قائمة قطاعات السكة */
rl,x,bx=FP-OFF (&a1[0] ); rg.es=FP-SEG (&a1[0]);
if(j= =2)
        {
        *(ukazad+3)=1;
```

```
gotoxy(1,25); printf(j=%d كود الخطأ =:d^M ,j,rl.h.ah);
*(ukazad+3)=2;
getch0;
}
                    /*محاولة قراءة القطاعات منزوعة التأطير */
                    /* absread0 رقم القطاع المعاب */
                         ويتم حسابة كالتالى:
nsek=rl.h.dh*360+rl.h.ch*9+ndefsek-10;
ndefsek            الرقم المتسلسل للقطاع
     tab             المعاب في الجدول
     ndefsek=2,   في هذه الحالة
     nsek=694*/
     for (g=1;g;g++)
     {
                    /* قراءة القطاع */
rl.h.dl=0;             /* الجهاز a */
rl.h.dh=1;             /* رقم الترويسة */
rl.h.ch=38;            /* رقم السكة */
char far *ukazad:
                    /* جدول السكة-القطاع-العنوان-القياس لكل قطاع */
chartabe[9][4]={38,1,1,2,38,1,2,1,38,1,13,2,38,1,4,2,38,1,5,2,38,
1,6 ,2,38,1,7,2,38,1,8,2,38,1,9,2};
    system («dir a:»)
    smes=pek (0,120);              /* جدول قاعدة الديسك */
```

```
ukazed=MK -FP(0,smes);
rl.h.dl=0;
rl.h.ah=0;
int86(0x13,&rl,&rl);
clrsc0;
rl.h.dl=0;                /* الجهازa */
rl.h.dh=1;                /* رقم الترويسة */
rl.h.ch=38;               /* رقم السكة */
rl.h.ah=5;                /* تابع التأطير العام */
rl.h.al=1;                /* عنوان جدول قطاعات السكة */
rl.x.bx=FP-OFF(&tab[0]);rg.es=FP-SEG(&tab[0[):
/* رقم القطاع المعاب */
ncek=rl.h.dh*360+rl.h.ch*9-10+2;
/* استدعاء 13h */
int86x(0x13,&rl,&rl,&rg);
/* طباعة كود الإنهاء */
gotoxy(10,9); print الإنهاء " =:d/ رقم القطاع المعاب= :d" .rl,h,ah,nsek);
/* التسجيل في القطاع المؤطر */
for (g=1; g; g++)
{
/* كتابة القطاع */
rl.h.cl=g;
rl.h.ah=2;                /تابع القراءة/
rl.h.al=1;
```

```
/* عنوان قائمة قطاعات السكة */
rl.x.bx=FP-OFF(&a[0[); rg.es= FP-SEG(&A[0]);
        if(j= =2)
        {
*(ukazad+3)=1:
        gotoxy(1,25);print                (محاولة إنهاء القراءة)
        {
memset?(a,0,100);
int86x(0x13,&rl,&rl,&rg);
/* استدعاء الإنقطاع 13h */
window(1,20,80,24); clrscr0; a[36]=0;
printf («كتابة=s« ,a);
window(1,1,80,25);
gotoxy(30,25); print(«j=%d كود الخطاء =;d« j,rl,n,ah);
*(ukazad+3)=2;
getch0;
}
printf( البرنامج أنهى العمل)
}
```

يمكن التنويع بالمؤشرات الموجودة في جدول مؤشرات الديسك في حال القيام بعمليات القراءة والطباعة، فمن المعروف بأن مجموعة copylock تحول الديسكيت إلى ديسكيت مفتاحي من خلال عملية التأطير العام للسكة ذات القيمة (0) ويتم القيام بذلك بتغيير بعض المؤشرات كمقياس القطاع، عدد القطاعات في السكة الواحدة وقيمة فاصل التأطير

(FORMAT). عملية التسجيل (الطباعة) / القراءة في التأطير اللامعياري، تنجز بمؤشرات معيارية. ولتبسيط العمل مع المثال 2.7 نورد أدناه وضعاً مختصراً لجدول قاعدة الديسك:

التداخل	المعنى
0	
1	يستخدم DMA
2	عدد الفواصل قبل فصل المحرك
3	مقياس القطاع
4	القطاع الأخير فى السكة
5	طول فاصل عملية القراءة/التسجيل
6	الطول الأعظمى للمعطيات
7	طول الفاصل لعملية التأطير
8	رمز المحمل فى حال التأطير
9	زمن الأنترفال الأساسى
10	زمن تشغيل المحرك

والإمكانية قائمة وكبيرة في أن تذيب في داخلها عملية تغيير المعلومات الوظيفية على الحوامل المغناطيسية.

سادساً – وضع معطيات في جسم البرنامج مرمزة:

تلوح وكأنها تؤدي نفس القيم والمعنى التي أرادها مؤلف البرنامج والمستخدم له والمباعة له رسمياً، ومنها ما يستخدم غالباً شكل السطر 2 و«فيو 2» والرقم المتسلسل في النظام للبرنامج، وهذه المعطيات تعطي الإمكانية لاحقاً لإستخدام طرائق الحماية القانونية، علماً أنها ليست بذات تأثير لدينا، أوردنا في القسم 2.1.1 (المثال 2.4) نموذجاً لتلك الحماية، توجد طريقتان عملياً للحفاظ على هذه المعلومات في حال إجراء نسخ غير مشروع للبرنامج هما:

• **التمويه**: تخزن المعلومة المراد إخفاؤها وعلى شكل مشابه للكود البرنامجي في ذلك المكان من البرنامج حيث تكثر الإنتقاءات، أو يتم تخزينها في سياق عمل بعض أجزاء البرنامج التي لا تقوم بالمعالجة في النظام الاعتيادي.

• **المقاومة النشطة**: تجري داخل البرنامج عملية تدقيق وبحث عن مدى وجود معلومات مخفية، وفي حال عدم كشفها يتم الانتقال للعمليات الجوابية، وأحياناً مجرد وجود مثل هذه المقاومة يعطي الإمكانية لإيجادها، وإختيار المعلومة مثال واضح على المقاومة النشطة وهي تعتبر بذاتها كوداً منفذاً جزئياً

2.2.3 الخصائص الفيزيائية – النفسية للإنسان المشغل:

١. خصائص مستخدم الضمانات البرنامجية:

هنا يمكن إيراد إمكانيات المستخدم وصفاته الإفرادية المميزة له، لذلك من الضروري استخدام أجهزة تسمح بتقبل واستقبال والتعامل مع خصائص الصوت، بصمات الأصابع.. الخ وعملياً فإن إدخال كلمة السر طريقة مناسبة لهذا الأسلوب. وفي المرجع (9,11) يمكن إيجاد معالجة واسعة لهذه المسألة إذا أمكن دخول البرنامج بتجنب وسائل التسجيل والإثبات والصوت والمراقبة للمستخدم فإنه يمكن أيضاً كتابة وإدخال «فيروس الحاسوب» التي تقلب كلمات السر وبسرعة كبيرة وكذلك نماذج التصور الرقمي للصوت أو لبصمات الأصابع... الخ، للوصول إلى المطلوب. وكما هو عادة فلا يتم التركيز كفاية على إمكانية الوصول غير المشروع للبرنامج عند حساب وسائل الحماية.

٢. الضغط النفسي على المستخدم:

ويكفي وبكل بساطة إخافته بالقول له أن المجموعة محمية، وإذا كنت تريد نزع الحماية فيمكن أن تتعرض المعلومات للتلف أو الفيروسات وهذا يترك أثره البالغ، ولايقوم المستخدم بعدها بإجراء أية تجارب على المجموعة.

2.3 تحليل سمات المحيط :

تعتبر وحدة مقارنة سمات المحيط (الوسط) إحـدى حلقـات الحمايـة، ولنزع الحماية لاضرورة لفهم تفاصيل منطقها بل القيام «بتصحيـح» بسـيط لنتائج المقارنة. لاتوجد في بعض المجموعات مثـل هـذه الوحـدة بـل يقـوم الإنسان بعملها، وخاصة في تلك المنظومـات الـتي تضـم معلومـات مرمـزة، حيث يقوم الشخص المشغل بالمقارنة في سياق التشغيل.

يمكن من الناحية العملية اختبار النماذج التالية للتحقق :

الأول: مجموعة من مشغلي المقارنة بين ماهو موجود مع ماهو مطلوب.

الثـاني: تجـدد توالـد الأوامـر المنفـذة بربطهـا مـع نتـائج عمـل آليـة الحماية.

مثلاً في الخانة الأولى يتم تخزين المحصلة الأوليـة المفتاحيـة التدقيقيـة لـ BIOS, في الخانة الثانيـة وفي سياق عمليـة الإنجـاز تُسجِّل المحصلـة الحسابية التدقيقية الناتجة كل المسألة الموضوعة.

جوهر وحدة المقاومة: طـرح قيـم الخانـة الثانيـة مـن الأولى، وإضافـة ناتج الطرح لكل خليـة في الذاكـرة العملياتيـة في مجـال DOS. إذا كـانت المحصلات لا تتطابق فلا يعود DOS قادراً على القيام بعمله.

الثالث: إنجاز عدد مـن العمليـات الحسـابية علـى المعطيـات وربطهـا بنتائج عمل آلية الحماية.

2.4 رد الفعل على الأعمال غير المشروعة:

وحدة الأعمال الجوابية تقوم بعمل جوابي قد يكون :

- «انتحار» البرنامج أو تخفيض جزئي لعدد الوظائف المنجزة.
- إعادة بناء غـير عكسـية للوسـط المحيـط ووفقـاً لاحتياجاتـه، وهـذا يعني أن الحاسوب لم يعد قادراً على العمل مع أيـة مجموعـة أخـرى غـير المحملة والموجودة.
- «السخرية» من المستخدم وذلك بتباطؤ العمليات الجوابية.

• تحطيم الوسط المحيط بما فيها تحميل وإضافة الفيروسات إلى الوسط البرنامجي وإيجاد «أحصنة طروادة» فيها.

ونذكّر غير المطلّع بأن «حصان طروادة» عبارة عن برنامج يقوم بتنفيذ أعمال ضارة لجهاز الحاسوب وللحوامل المغناطيسية، تكمن خاصيته الأساسية في أن إطلاقه للتنفيذ ويعني قيامه بالتحطيم أو إخراج الملفات الخاصة وكشفها يجب أن يتم من قبل المستخدم. ولهذا البرنامج تسمية ولقب يحدده مالكه الجديد ويطلق في التنفيذ لمعرفة هدفه وتوجهه. يتميّز «حصان طروادة» ببساطة التكوين. اكتبوا برنامجاً مؤلفاً من مشغلين من أجل أبعاد المعطيات، و كوّنوا ملف EXE ومن ثم قوموا بتسجيل وتخزين الملف الناتج في نموذج الحماية الموجود لديكم وفي حال الإخلال بالشروط قوموا بالنسخ مباشرة من برنامج، «حصان طروادة» من الكاتالوج الرئيسي؛ وكونوا على ثقة بأنه لن يمر شهر إلا ويطلق أحد ما هذه «المسألة الجديدة». يمكن لقسم العمليات الجوابية أن يقوم بنزع تمويه منظومة الحماية، وللتخفيض من قيمة «عامل نزع التمويه» يتم استخدام أفكار حدية للعمليات الجوابية وذلك بعد عدة مرات من الإطلاق وتأجيل الانتقام، أي أن العملية الجوابية تجري بعد فترة من الزمن، أو خلال عمل جزء من البرنامج والمؤلف من قسم مقارنة سمات المحيط. تملك فكرة ومبدأ تصاعدية العملية الجوابية وتدرجها وصولاً حتى توقف القدرة على العمل بشكل كامل بعض الإيجابيات في مخطط الحماية :

• لايتبين مستخدم النسخة غير المشروعة مباشرة.

• تظهر البرنامج المستخدم وكأنه موصول ومترابط بشكل سيء مما يبعد التفكير في نزع الحماية.

• تصيب المستخدم بالخوف الكامن في إمكانية الحصول على عملية مجهولة وغير معروفة.

• النواحي السلبية لهذه الطريقة :

* ظهور علائم مضادة، ذلك لأن العملية الجوابية يتم تقبلها وكأنها خاصية المنتج ذاته ولكن في شكل غير محمي. يزداد مردود العملية

الجوابية إذا كتبت بطريقة مناسبة في الوثائق، ومن الممكن عدم وصف جميع العمليات الجوابية والاكتفاء ببعض منها.

لنتحدث قليلاً عن مهرجان (استعراض) الاحتمالات:

يجب تجنب وضع حدوث تشكيل ذلك الاستعراض، بواسطة وضع قاطع في قسم العمليات الجوابية، لأنها تؤدي لضياع جزئي لوظائف الحاسوب، ويبقي مكانها برنامج خاص، بالإضافة إلى بقية الوظائف التي لايمكنه فيزيائياً تنفيذها. إضافة لذلك يجب ألا تتواجد كل آليات الحماية ولا حتى التنويه عنها.

الفصل الثالث

تنظيم الحماية من النسخ والتقليد غير المشروع

3.1 برمجة أنظمة الحماية:

لنلاحظ نقطتين تقنيتين يمكن لهما أن تؤثرا على برمجة وسائل الحماية:

يمكن وضع الحماية في حزمة (مجموعة) المنظومة أو أن يكون لها مقراً خاصاً للإقامة في المجموعة، ويمكن للمقر أن ينفذ مختلف الوظائف: عرقلة كشف نظام الحماية، حماية المصورات، حماية أقسام الديسك. وغالباً ما يكون المقر على شكل مونيتور، منفذاً للترميز وفكه، (المثال 3.1)، وفي المرجع [6] أورد المؤلف نموذجاً لتحقيق الحماية مع استخدام المقر. أما الحماية بدون مقر فهي أكثر بساطة ويسر في التحضير والتصنيع، ولكنها أكثر قابلية وسهولة لتجاوزها وتجنبها. وليس ممكناً بشكل دائم استبدال نموذج وسائل الحماية بآخر، كي تصبح دائماً عملية متابعة وملاحقة برنامجكم أكثر صعوبة، عليكم أن تدخلوا وتخزنوا في الذاكرة قبل المنافس المحتمل، وهذا يخص الأنظمة ذات الحماية مع قسم المقر.

يمكن تصوّر ترتيب العمل بشكل مبسط جداً لنموذج يعمل في حال التحميل كما يلي:

المحمل، MBR – القطاع، قطاعBOOT ، IBM B10.COM6، IBM، DOS.COM، COMMAND.COM، والأقسام المكونة لـ CONFIG.SYS والأقسام المكونة لـ AUTOEXEC.BAT.

يمكن للنظام مع قسم الإقامة المجهز به، أن يتغلب على محاولات الحصول من الحاسوب بنموذجك الخاص وبمرحلة مبكرة ومن ثم القيام بتنظيمه للعمل بشكل معياري.

١. الحماية إما أن تكون ملتحمة أو مبنية:

في الحالة الأولى فإن الحماية هي عمل النموذج المحقق لها ولا تتقاطع زمنياً مع عمل النموذج الوظيفي، يُوضع في البداية نموذج الحماية، ومن ثم الموديل الوظيفي. ويمكن متابعة لحظة نقل التحكم من أحد النماذج إلى الآخر، وتجنبه نظام الحماية دون تحليله. الكثير من المجموعات التجارية لترتيبات الحماية قائمة على هذا المبدأ، ونوع الحماية الذي يجلب معه معالجة أولية للملف المحمي، وبحيث لايمكن أن تشاهد فيه نمطاً (نموذجاً) مشابهاً للأولي.

أما الحماية المبنية، فيتم عملها قبل وخلال عمل النموذج الوظيفي، ولنزع مثل هذه الحماية من الضروري بذل جهد أكبر.

تكمن أسهل طريقة تسمح بوضع حماية من النوع الأول، ومشابهة للثانية في جميع أجزاء البرنامج المحمي وأجزاء DOS في نموذج الحماية، والنموذج المحمي. إذا أضفنا للكود المحمي ثابتاً ما محدداً، فيصبح الأمر وكأنه مجموعة جديدة وغير معروفة تماماً. يجب إعارة انتباه خاص لحماية وسائل الحماية ذاتها.

يمكن أن يعطي تحليل محتواها، ومتابعة عملها، معلومات لاتقدر بثمن لإمكانيات نزع وكشف البرامج المحمية من خلالها وبمساعدتها. إذا كانت تلك البرامج ذات انتشار تجاري واسع، فيجب أن تمتلك إمكانية لتشكيل عدد محدد من نماذج الحماية، وأن تدمر ذاتها بذاتها.

قبل تصميم وسائل الحماية من النسخ غير المشروع يجب تصور تنظيم استخدامها، يوجد اختياران في هذا المجال:

- المستخدم يضع الحماية بذاته.
- يضع الحماية من الشركة التي قدمتها.

أفضليات الخيار الأول تكمن في إمكانيات النسخ الواسع للوسائل المصنعة. والثاني إنجاز عمل أكثر نوعية ودقة واسترداد وتماسك أمام محاولات تفكيك وكشف وسائل الحماية.

أي خيار يجب اعتماده؟ لا يمكن تقديم وصفة عامة، ويمكن النظر بكل حالة على حدة، في حال اعتماد الخيار الأول: يجب معالجة مسائل محددة: مثل في أي ناقل ستنتشر منظومة الحماية (الديسك، الشريط ..الخ) وكيف سيتم تركيبها؟. وتكمن الصعوبة في ضمان الاحتمال الأدنى لإمكانية كشف أسرار «الشركة»، وأنظمة الحماية، عادة تكتب بنفس لغة النموذج المحمي، أو تستخدم لهذا الغرض جاهزة بلغة Assembler.

الحمايات من نوع تستخدم لغة Assembler، التي تسمح بسهولة التعامل مع التجهيزات واستخدام وظائف التوابع المعيارية. إلا أنه لهذه اللغة بعض النقائص أهمها:

• البطء ،والجهد الكبير الواجب بذله لكتابة اينترفيس متنوع ومتماسك مع الأدابتر وعدد من الوظائف المعيارية.

• صغرالنموذج المعبر عن النتائج، والذي يمكن بسهولة تفكيكه وإخضاعه.

بإمكان لغات المستوى العالي أن تتجاوز هذه النقائص. لذلك تستخدم عادة توليفة مؤلفة من Assembler مع لغات (سي، باسكال ..الخ) على المصمم أن يحفظ في جعبته المعدات الضرورية وأن يعمل لإيجاد الحماية المناسبة بسرعة كافية وألا ينشغل عن الأمر الأساسي في عمله، وأن تكون لديه مكتبة إضافية للغة التي يستخدمها في البرمجة تحوي مجموعة من ملاحق البرامج، والبرامج الثانوية والخاصة. على أن تكون النماذج فيها مقسمة وظائفياً على ثلاثة أقسام.

نورد أدناه تركيباً محتملاً لهذه الأقسام بالاعتماد على لغة «C».

3.1.1 قسم الحماية من «المجرد»:

• حساب المحصلة الكونترولية للملف:

int ksum file (char * imj) حيث imj اسم الملف.

البرنامج الثانوي يعيد المحصلة الكونترولية لجميع كلمات الملف المشار إليه أو يظهر (0) في حالة النهاية الطارئة.

• حساب المحصلة الكونترولية للمسألة أو أقسامها في الذاكرة العملياتية:

int ksumpat (int M)

البرنامج الثانوي يعيد المحصلة التدقيقية لجميع كلمات (N) المقاطع القريبة من الذاكرة العملياتية أو يظهر (0) في حالة النهاية الطارئة.

• تدقيق الذاكرة الحرة للحاسوب: ويمكن إنجاز ذلك بمساعدة التوابع المكتبية المعيارية للغة ذاتها.

• البرنامج الثانوي «توافه»:

int pusto (inti) ويحوي مجموعة من الأوامر المتجددة بذاتها.

إذا كان مؤشر الدخل I=0 يتم تنفيذ الأوامر غير الخطرة والتي تؤشر على البرنامج. إذا كان (io) وأوامر غير متوقعة، ويؤدي تنفيذها لتنفيذ عمل خاطئ في البرنامج، فتنهاركامل المنظومة.

• برنامج تدقيق زمن التنفيذ:

int timexe (int t)

يعيد (0) إذا كان الفاصل بعد آخر تنفيذ لا أكثر من (1) ثانية وإلا فإنه يضيع في الذاكرة العملياتية للمسألة المنفذة.

3.1.2 وحدة مواصفات المحيط:

التشكيل الخاص للديسكيت:

int form disa (int k, *nom, shan*str, mtk)

حيث k – عدد القطاعات التي يجب تأطيرها بصورة خاصة.

str – السطر، الذي يجب كتابة محتواه في K.

ki – مقياس القطاع لكتابة stan.

nom – مؤشرات للتأطير.

الهيكلية :

nom : int   nom [ ] [4]

حيث :

nom [ ] [4] – رقم السكة

nom [ ] [1] – رقم

nom [ ] [2]

رقم القطاع :

(1-9) أرقام صحيحة.

-9 أرقام غير موجودة.

nom [ ] [3]

كود العملية :

مقياس القطاع.(3-0)

المقياس المعياري.2-

البرنامج الثانوي يعيد محصلة «كود» الإنهاء إلى القراءة وإلى القطاعات.

- قراءة مجموعة القطاعات وحاصل كود الإنهاء:

int wossdisa (int k, int*nom)

الإعادة لجميع القطاعات.

- حساب المحصلة التدقيقية :

int ksum o وإعادتها.

- تحديد مكان توضع الملف على الناقل :

.Long mesto file (chan*imj)

حيث imj اسم الملف. البرنامج يعيد عنوان مكان الملف الفيزيائي إلى الناقل.

• مراقبة قائمة التجهيزات int oborua o .

• الكتابة في مكان محدد في الملف.

int Zahesi (char*imj), I6 char*str,i1)

I – مكان في الملف.

str – سطر مكتوب في الملف مع تسمية بداية، str إعتباراً من K بايت.

I1 – عدد الخانة للسطر، والتي يجب أن تلحظ في الملف.

البرنامج يعيد (1) في حال الطوارئ و(0) في حال الإنهاء الطبيعي.

• الكتابة في الملف (التسجيل) بعد السطر المحدد:

int Zanesik (char*imj, char*str, i ,char*nz)

حيث imj – اسم الملف.

str – سطر الملف بديل الذي سيسجل.

I – خانة من السطر nz.

j – عدد خانات السطر nz.

nz – سطر للتسجيل في الملف.

3.1.3 قسم العملية الجوابية:

• كتابة وتسجيل في مكان ما عارض في الملفات المحددة عدد ما من الخانات ذات الأرقام (0):

int filebait (char*ust, char dir, char file, int n)

حيث:

ust – جهاز. إذا وجدت إشارة * فهي خارجية التجهيزات.

dir – سكة. إذا وجدت إشارة * فيعني كامل الكاتالوج.

file – اسم الملف والتوسع.   في حال * فيعني كل شيء.

n – العدد الأعظمي الممكن للخانات الأصغرية المسجلة.

• عمل الشاشة :   ekr bait 0

العمل مشابه للظاهرة الناجمة عن ضياع الوصل والربط الفيزيائي ما بين Display والآلة الحاسبة الألكترونية.

• تشبيه الدخول في rezonans القطع الميكانيكية للآلة الحاسبة الألكترونية :

rezonans (char*ustr)

حيث :

ustr – معدات خارجية بما فيها الطابعة (LPT:)

• تشكيل «حصان طروادة» :

int konia (char*imj, char*str)

حيث :

imj – اسم الملف المشكل «حصان طروادة».

str – بوفر في ذاكرة المسألة وحيث يتم حفظ الملف المشكل.

ملحق البرنامج يعيد (1) إذا كان لحصان مؤلف بشكل ناجح و(0) في الحالة المعاكسة.

• نسخ إضافة الملفات EXE:

int execopy (char*imj1, char imj2)

حيث :

imj1 – اسم الملف EXE.

imj2 – اسم الملف EXE.

ينسخ ملحق البرنامج الملف EXE مع الأسم [ ]imj2 في الملف ذو الأسم [ ]imj1. يعيد ملحق البرنامج إلى (0) في حال الإنتهاء والإنجاز والطبيعي وإلا فيعود إلى (1).

• نسخ الملفات COM.

```
int comcopy (char*imj, char*imj2)
```

حيث:

imj1 – اسم الملف COM.

imj2 – اسم الملف COM.

ينسخ ملف البرنامج الملف COM مع الاسم [ ] imj2 في الملف ذو الاسم [ ] imj2. اسم الملف الناتج [ ]imj1. يعود ملحق البرنامج إلى (0) في حال الإنجاز الطبيعي، أو إلى (1) في الحالة المعاكسة.

• نسخ ملفات EXE, COM:

```
int eccopy (char*imj, char*imj2)
```

حيث:

imj1 – اسم الملف EXE.

imj2 – اسم الملف COM.

ينسخ ملحق البرنامج (الملف COM) ذو الاسم [ ]imj2 في الملف []imj1 يسمى الملف الناتج [ ]imj. يعود الملحق إلى (0) في حال الإنجاز الطبيعي أو إلى (1) في الحالة المعاكسة.

3.2 حماية الملفات الكتابية والمصورات:

يتطلب ذلك عملاً وجهداً ومشقة أكثر من حماية البرامج ذاتها، وللأسباب التالية:

• هذه الملفات عادة غير منجزة تماماً. لذلك يجب استخدامها بصورة مشتركة مع وسائل الحماية.

• لها قينونة احصائية معروفة.

يمكن استخدام الأساليب التالية لحماية هذا النوع من الملفات:

١. ترميز الملفات (من الممكن مع توليفة مختصرة).

٢. إيجاد مونيتور مع مقر إقامة.

٣. إيجاد برنامج يضم ملفات الكتابة والمصورات.

ومع ذلك تبقى قائمة مسألة النسخ غير المشروع للمصورات أو النصوص مباشرة من الشاشة، وذلك قبل نسخ البرنامج الموضوع في الذاكرة. لتجنب ذلك يجب امتلاك BIOS غير معياري أو DOS لامعياري. أو تقوم بنفسك بمعالجة جميع الإنقطاعات الناجمة عن الضغط على الزر. وبذلك تحفظ وتبقى تحت المراقبة كل التجهيزات الخارجية. تسمح بعض أنظمة حماية الملفات الكتابية بطباعتها فقط دون استعراضها على الشاشة. وهذا يعني أن الملف المحمي يحوي Driver لجهاز الطباعة. وهذه الطريقة ترفع درجة حماية الملف.

نورد فيما يلي مثالاً لعمل المونيتور مع مقر الإقامة، ومبادئ عمله، وهي كما يلي:

- المونيتور يأخذ لنفسه المؤشر (12h) ويتابع ويلاحق وظيفة «القراءة».
- يدقق ما تم حسابه في البوفر. وإذا كان النص مرمزاً، فيتم تفكيك الرمز في الذاكرة العملياتية.
- ولكن العمل مع (12h) ليس ملائماً دائماً ذلك لأن جميع أنواع «المجرد» المقيم تلاحق عمل (13h) و(21h). لذلك ينصح بإجراء العملية الإضافية التالية:

- وجه المونيتور إلى أي مؤشر آخر مثلاً 80h.
- قلّد محتوى 21h في العنوان 80h.
- صحح في البرنامج المستخدم التوجه من (21h) إلى (80h).

ولهذا الغرض يكفي أن تستبدل في كامل الملف الكود CD21h بـ CD80h. عندها يظهر المونيتور المقيم بلغة Assembler كما يلي المثال 3.1:

المثال 3.1:

```
sseg segment stack
        db 256 dup(?)
sseg ends
cseg segment
assyme cs:cseg,ss,sseg
z       proc far
        push ds
        mov ax,0
        push ax

                push cs
                pop ds
                mov dx,offset prod
                movax,25h
                mov al,80h
                int 21h
```

الخروج وبقاء المقر

```
        mov ax,100
        mov al,0
        mov ah,31h
        int21h
```

ret

prod:cmp ah,3fh: وظيفة القراءة:

jne a0 لا :

int 21h نعم :

push ax تذكر: Register

push cx

push di

push bx

push dx

pop bx

إذا كان الرمز الأول، فيجب تفكيك رمز السطر الأول:

mov ah,ds:[bx] هل يجب تفكيك الرمز أم لا؟:

cmp ah,*

jne aa لا :

mov di,0 نعم :

a1: mov ah,ds:[bx+di]

قسم تفكيك الرمز:

فى هذه الحالة فإن جميع الرموز عدا الأول تبقى بدون تغيير:

add ah,0.

نهاية قسم فك الرمز:

mov ds:[bx+di],ah

inc di

loop a1

```
        mov ah,
        mov ds:[bx],ah
aa:pop bx;                    إرجاع Register
    pop di
    pop cx
    pop ax
    iret
a0:int 21h
    iret
cseg ands
and z
```

3.3 التأثير المتبادل لمنظومات الحماية مع الوسط المحيط:

3.3.1 التأثير المتبادل مع الفيروسات:

قد يؤدي ذلك التأثير المتبادل إلى نتائج غير محبذة، حيث تساعد الفيروسات في نزع منظومة الحماية، كما أن نظام الحماية ذاته يحمي الفيروس من البرامج المضادة، وللنظر في الأمر:

- الفيروس يساعد في نزع حماية المنظومة:

لنفترض أن منظومة حماية الديسك تضم المحمل المسجل عوضاً عن القطاع BOOT النظامي للديسك، إضافة للوظائف التقليدية الضامنة لتدقيق كلمة السر عند التعامل مع الديسك. يمكن إعادة تأطير وتوجيه وتشكيل القطاع BOOT من أجل عدم السماح بقيام العمليات البسيطة للقراءة أو التسجيل. وهكذا يتم إبعاد تدقيق كلمة السر. وقد لا يكون القطاع BOOT كما يشتهي الفيروس، عندها لاتتم العدوى. عند محاولة نقل العدوى للنواقل المحمية تظهر معلومة عن ذلك، تنتقل للمنظومة العملياتية

وهذا يعتبر كأحد العلائم المبنية عن تواجد الفيروس. لا تستطيع الفيروسات الحديثة نقل العدوى لديسك محمي جيداً وذلك للأسباب التالية:-

ينتشر الفيروس عادة مع نسخ البرامج غير المحمية، وهجوم الفيروس ليس كبيراً ولذلك من الصعوبة إيجاد لوغاريتم يتجاوز عقبات الحماية المعقدة. وإذا كانت الحماية سهلة التجاوز فيجب تغييرها.

• الفيروس ينزع بصورة كاملة لا عودة عنها منظومة الحماية:

نفترض أن حماية الملف قائمة على أساس تغيير بعض المعطيات على الديسك أو داخل الملف، في حال العدوى فإن قسم من الفيروس يمكن أن يتوضع في تلك الأماكن من الديسك، أو الملف وبذلك يحمي المعطيات الموجودة هناك. بعد ذلك يصبح العمل مع الملف المحمي غير ممكن. ذلك لأن الحماية ستعتبر أن ما يجري هو عبارة عن نسخ غير مشروع وليس عمل المنظومة ذاتها. ومثل ذلك يمكن أن ينشئ إذا كان توجه الفيروس نحو الضرر ولمحي بعض مناطق الديسك.

عموماً ينصح بتدقيق عمل النماذج المحمية المتوضعة وبعد تنظيف الديسك من الفيروسات.

تنشأ أكثر الحالات جدية عندما يستوطن الفيروس في البرنامج المحمي من «المجرد»، ويمتلك قسماً للعمليات الجوابية. يلاحظ البرنامج المحمي المصاب بالعدوى ما يجري، ولكنه يقيم الأحداث باعتبارها محاولة لنزع الحماية. مما يؤدي لإعادة معالجة وظائفه في قسم العمليات الجوابية. وهذه الحالة خطيرة جداً. وتعني أن الفيروس قد تمكن تماماً من المنظومة التي في ردها تنقل كل المعطيات إلى الناقل المغناطيسي متجاوبة مع متطلبات وإجراءات الفيروس. ولايبقى للمستخدم الذي اشترى البرنامج غير حزنه وإيقاف المنظومة.

ومن هنا نخرج باستنتاج هام وهو أن العملية الجوابية يجب ألا تكون حادة في الاتجاه الصحيح وغير مضرة.

• نظام الحماية يجنب الفيروس من البرامج المضادة للفيروسات:

يوجد خياران ممكنان وفقاً لأنواع الفيروسات:

١. نفترض أن الديسك المصاب بالفيروس أثناء التحميل، تمت معالجتة من قبل برنامج الحماية، وبحيث يصبح من الصعوبة بمكان التسجيل في MBR أو القطاع BOOT.

٢. نفترض أن الملف المصاب محمي من التقليد والنسخ، وذلك بالقيام بجعل كتلته متحولة في الوضع العادي، وتوجيهها نحو تحقيق تحول معاكس قبل التنفيذ مباشرة.

في الحالة الأولى، لايمكن معالجة الديسك المصاب دون إعادة تسجيل القطاعات.

في الحالة الثانية، لا يمكن إكتشاف الفيروس قبل استدعاء الملف للتنفيذ وكذلك معالجة الملف المصاب. وهذا يعني:

* ضرورة إجراء تنظيف دقيق للمنتج المحمي من الفيروسات قبل استخدام وسائل الحماية.

* قبل المعالجة من الفيروسات يفضل نزع نماذج الحماية إذا أمكن.

3.3.2 التأثير المتبادل لأنظمة الحماية والبرامج المضادة للفيروسات:

تقوم بعض البرامج وغالباً المضادة للفيروسات، بتدقيق النواقل، الذاكرة العملياتيه، المدخلات اللامعيارية والعمليات المشابهة للفيروسات.

يمكن أن تستخدم وسائل الحماية طرائق عمل مشابهة ومماثلة للفيروسات وتضع معطياتها أيضاً هناك حيث تتواجد معطيات الفيروسات ذاتها.. ولذلك فإن الإحتمال قائم بألا تقوم البرامج المضادة للفيروسات بالعمل بوجود نماذج حماية، أو أن نجد الفيروسات هناك، حيث توجد نماذج حمائية.

نصائح لتصميم أو اختيار نظم الحماية:

• عدم استخدام طرائق البرمجة الواسمة للفيروسات، وكذلك مجالات الذاكرة التي تستخدمها الفيروسات عادة.

• تجنب عمل البرامج المضادة للفيروسات في الوسط المحمي.

• توثيق كل الحالات الخاضعة للشك بدقة متناهية.

3.3.3 التأثير المتبادل «لفيروس» الحاسوب مع نظام الحماية:

«فيروس الحاسوب» عبارة عن برنامج خاص موجه لتجاوز مختلف أنواع الحمايات. وخلافاً للفيروسات تنقسم ذاتياً بينما تتطلب الفيروسات وجود ناقل: ديسك أو ملف، ولا تملك تحديداً من حيث تزايدها وقياسها. وإذا لم تكن وظائف الإنقسام ضرورية لنزع الحماية، فلا يمكن عندها تمييز «الفيروس» عن البرنامج العادي. وهي ذات تخصص ضيق، يمكن القضاء عليها من خلال نوعٍ محدد من الحماية. عملياً الحماية من «فيروس الحاسوب» مرتبطة أيضاً مع الحماية من تأثير البرنامج ذاته. ويمكن إيجاد سلسلة كاملة من أنواع «الفيروس» يتخصص كل منها بجانب أو بعض جوانب الحماية. ويحقق استخدام «فيروس الحاسوب» من حيث طريقة «الإنسان – الآلة» في نزع الحماية. وحيث توجد عمليات نمطية أحادية الطابع يمكن مصادفة «فيروس الحاسوب»

3.3.4 التأثير المتبادل لمختلف أنظمة الحماية:

يمكن مواجهة ثلاثة احتمالات تؤثر على مدى النجاح في تصميم ووضع أنظمة الحماية وتحقيقها:

١. ناقل المنتج المحمي «الديسك، الحاسوب» مخصص بالكامل لعمل هذا المنتوج. وتتحدد إمكانية الحماية بدرجة تخصصية ومهارة واضعها.

٢. استخدام سلسلة من المنتجات غير المحمية على الناقل إلى جانب المنتج المحمي.

٣. وجود عدد من المنتجات المحمية على الناقل.

يتزايد احتمال تأثير الحماية المتبادل. ويجب اتخاذ الإجراءات اللازمة لتجنب تأثير أحد المنتجات على غيره وبالعكس، خاصة إذا كانت جميعها موضوعة وفق مبدأ واحد. نادراً ما يهتم المصمم لنظام حماية محدد بمسألة تجانس هذا النظام مع الأنظمة الأخرى. مما يؤدي إلى عدم إنسجام

أنظمة الحماية حتى مع ذاتها، مما يقلص من إمكانيات استخدامها. وعلى سبيل المثال لا يمكنك أحياناً أن تحفظ في ديسك واحد ملفين محميين بنفس الطريقة.

3.4 إختيار وسائط الحماية:

وهكذا فأي طريقة حماية نستخدم؟

نورد بعض الأفكار حول هذه المسألة. من المفيد استخدام تلك الحماية التي يكلف نزعها أكثر من مجموع تكاليف وضعها، مضافاً إليها مقدار الخسائر (الأرباح المفقودة) الناتجة جراء وصول أشخاص آخرين للمعلومات المحمية. تتحدد طريقة الحماية بالمعلومة المحمية ذاتها. والحماية يجب أن تكون معقدة بمقدار إنتشار نسخ المنتج المحمي وبطء تغيرها.

وفي كل الأحوال من المفيد استخدام منظومة مركبة للحماية، تجمع الأساليب الحقوقية والتنظيمية والجهازية والبرامجية. يجب أن تخضع الحماية لإعادة نظر، وأن تتبدل ديناميكياً وذهنياً إذا وجد تهديد لكشفها، أو إذا تغيرت قيمة المعلومة المحمية. كما يجب أن يكون المنتج المحمي بسيطاً لدرجة كافية حين الإستخدام، وألا تؤثر آليات الحماية على تقنية العمل الروتينية للمجموعة المراد حمايتها.

ومن هنا يُنصح بإستخدام طرائق متشاركة متعددة ومتنوعة ومركبة لتقوية عامل الحماية ذاته.

3.4.1 مقارنة مختلف أنظمة الحماية من النسخ:

نفتقد في الوقت الحاضر – من حيث الموثوقية – لنقاط مقارنة بين مختلف برامج الحماية، وتتم المقارنة تجريبياً، ويلعب دوراً هاماً في ذلك مدى تأهيل الشخص المقارن. وتحمل جميع المقارنات للأسف طابعاً نوعياً وغالباً ذاتياً. عند إختيار نظام الحماية يجب ملاحظة بعض العوامل الهامة ومنها:

- الوظيفة المنفذة.
- القدرة على مقاومة جهود نزع الحماية.

• دقة التجهيز وغياب الأخطاء.

• التجانس مع البرامج الأخرى.

• سهولة الاستعمال.

لننظر في تلك العوامل باختصار. وللتبسيط نقسم الوظائف المنفذة لعدة مجموعات:

أ. توجه آليات الحماية إلى الملف أم الديسك؟

في الحالة الأولى، فإن التبديل في الملف أو الديسك، موجه لحماية الملفات المتشابهة والمتماثلة (الحماية من النسخ غير المشروع). في الحالة الثانية، فإن التبديل موجه لحماية مجالات الديسك أو مجموعة كبرى من الملفات (من النسخ غير المشروع). توجد حلول وسطية ومركبة من الأمرين.

ب. على أي ناقل يجب أن يتواجد النموذج المحمي؟

علماً بأن أنواع النواقل معروفة ومنها الديسك، التصوير، الشريط.. الخ.

ت. أي الملفات يمكن حمايتها؟

(قاعدة المعلومات، ملفات النصوص، الأشكال البيانية.. الخ).

ث. إستخدام تجهيزات خاصة اضافية لتحقيق الآلية الحمائية.

ج. خصائص الوسط المحيط المستخدمة في الحماية (أقسام الحاسوب التي لايمكن نزعها، تجهيزات خاصة، ديسك، خصائص الوسط البرنامجي، المعلومات التي يدخلها المشغل.. الخ).

ح. إمكانية التوضع في منظومات برامجية أخرى.

القدرة على مقاومة جهود نزع الحماية:

تنص الملاحق المصاحبة لمختلف أنواع المجموعات على أن تعقيد نزع الحماية وصعوبتها، من حيث كمية العمل المطلوبة، يمكن مقارنتها مع متطلبات كتابة مجموعة برامج كبيرة. وهذا ينطبق عملياً على المحاولة الأولى لنزع الحماية فقط، وتصبح المسألة أمراً عادياً في المحاولات التالية،

وفي المجموعات الأخرى المماثلة، يجب أن يؤخذ بالحسبان تقادم وسائل الحماية أيضاً، والمرتبط مع التعرف التدريجي لدائرة واسعة من المبرمجين مع طرائق الحماية.

يمكن مقاومة التقادم من خلال استخدام طرائق معيقة لإعادة وضع البرنامج الأولي وعلى أساس تحليل الجانب المحمي، وذلك بالقيام بتغيير متواتر لطرق الحماية، وبتصميم وسائل متبدلة ومجهزة وفقاً لتوسع دائرة المستخدمين. وفي هذا المجال أيضاً تتاح إمكانية مقاومة الوسائل المعيارية والواسعة الإنتشار لنزع الحماية. على البرنامج المحمي أن يعمل بسوية واحدة قبل وبعد استخدام وسائل الحماية. هذه الوسائل التي تعقدت في الفترة الأخيرة.

في سياق وضع نموذج الحماية تدخل تبدلات وتحولات قيمة على مخططه الأولي. وهذا يتطلب مقدرة نظرية كافية إلى جانب الإثباتات العملية أيضاً.

التجانس والتشارك مع باقي البرامج:

عامل غير ملزم، لكنه يوجد السهولة لعمل البرنامج المحمي في مختلف الظروف العملياتية ويوسع من آفاق استخدامه. العمومية الكبرى نختلق متاعب لنزع الحماية. وتعتبر حساسية وسائل الحماية نحو التجهيزات والوسائل المستخدمة والقائمة بعمليات لامعيارية النقيصة العامة المشتركة لها. ليس ضرورياً أن يكون لنظام حماية محدد لمنظومة عملياتية محددة قابلية عمل مع نوع آخر، أو نمط آخر لمنظومة عملياتية ثانية. كما أن سهولة الاستخدام والعرض الجيد والتعامل مع المستخدم توسع من أفق انتشار وسائل الحماية وتحقيق كامل إمكانياتها.

كما أن بعض وظائف الصيانة يمكن أن تشكل عاملاً مناسباً ليس للمستخدم فقط بل المتعدي أيضاً. مثلاً إذا وجدت وظائف نزع الحماية في المجموعة إضافة لوظائف حمايتها، فتدقيق مؤشرات الحماية وباقي الوظائف يمكن أن يؤدي لكشف أسرار البرنامج، مما يستوجب ضرورة النظر بالإجراءات المعاكسة المناسبة. بعض الهيئات تلجأ لوضع

أنظمة حماية خاصة بها بلغة Assembler وهي غالباً ليست أفضل من المعيارية.

3.4.2 مواصفات الحماية الناجحة:

١. ألا يمكن تجنبها بسهولة وبأساليب علنية، مثلاً الطراز المبكر.

watch DOG.Segurity system يمكن ببساطة تجنبه.

٢. ألا تكون طرائق نزعها معروفة بشكل واسع. وعلى سبيل المثال في منظومة Advanced disk managem.system يمكن ذلك، والمرجع [12] يرى بأنه في حال إجراء تغيير (ADM.SYS.V.1.00) وبتداخلها مع (OAF7h:75h-EBh,05BBh:13h-0Ah) فإن البرنامج سيسمح بالمرور لأي كلمة سر كانت. في حال كتابة (0) في القطاع 2 (العمود (0)، الجهة (0)) وفقاً للتداخل 01F1h والADM.SYS Driver غير محمل، تنقطع الحماية.

٣. تغير القطاع MBR وبرنامج NDD. من السهولة عندها فصل البرنامج ADM.

عملياً فإن مجموعة ADM ليست سيئة ولكنها منظومة متقادمة تسمح بوصول محدد للديسك، وهي معروفة وتلعب أيضاً دوراً مساعداً في نزع الحماية وموجودة في العديد من البرامج: ADM ,SAFER ,SADM ,ADMBRK,... وأخرى.

لا تؤمن بعض المنظومات العملياتية مثل DR-DOS V5.0 حماية أقسام وفصول الديسك.

NDD يلعب دوراً مساعداً بشكل جيد في حال منظومات (PC-Vavit 4.2) سابق (pc-Lock) و(secretLock/unLock).

3.4.3 وسائل الحماية المعيارية:

بعض المجموعات المشهورة (PC TOOLS, NORTON) تضم نماذج تسمح بحماية الملفات والديسك من استخدامها غير المشروع والنسخ، وتستخدم لذلك وسائل الترميز وكلمات السر. ونقيصتها الأساسية في سعة انتشارها، ونسخها وتقليدها، غير المأذون به، مما يعطي مجالاً واسعاً

للبحث عن طرائق للعمل المعاكس لهذه الأساليب الحمائية، وهي لا تستخدم في المنظومات التجارية. ونلاحظ هنا مجالين لاستخدامها:

١. الحماية من الزملاء متوسطي الكفاءة، الساعين لإستخدام نتائج عملكم.

٢. الحماية من الرؤساء والإداريين، الذين يمنعون موظفيهم من استخدام بعض أنواع البرامج في الحاسوب (الألعاب مثلاً).

3.5 بعض وسائل الحماية الخاصة:

نعرض فيما يلي بعض المجموعات لوضع آليات حمائية على المنتجات البرنامجية، هذا العرض ليس كاملاً ولا يطمح ليكون كذلك:

١. 25CHECK, VER 2.03, (C)25 SOFT 9.2.91

وهي عبارة عن مجموعة برامج تقوم بالتدقيق على كمال ترتيب وبناء بعض المواقع (الملفات، القطاع MBR، مقاطع BOOT، المواقع في ملفات المعطيات) وذلك باستخدام Cyclic Redundanc code، وحساب المحصلات التدقيقية على أساسها.

٢. **EASYPROT, VER2.0** ويتألف من النماذج التالية: (INSTAL, TRANSFER) وهو عبارة عن نظام حماية المجموعات البرنامجية، موضوع على أساس نموذج توربو – باسكال 4.0 وأكثر، للحماية من النسخ غير النظامي. يُنتج عدة نماذج منه.

٣. Copylock. Comii, (C)Link Computer 1988

تتمتع حماية Copylock بسمعة جيدة، باعتبارها قد تحولت لنوع من المعيار في مجال (دي فاكت).

والعديد من التصاميم اللاحقة تفيد ببساطة نص هذا البرنامج.

تستخدم في سبيل الحماية مشكلة بطريقة خاصة. يقوم copylock بجعله مفتاحياً بتغيير ترقيم القطاع الأخير في السكة إلى (30) بدلاً من (9)، وطول المقطع بالنسبة لعملية من (50h) إلى (15h). ومن ثم تتأطر السكة رقم (12) وبمؤشرات معيارية. وفي هذه الحالة فإن القراءة الأولية

قبل format للسكة (0)، – والنص اللاحق عليها – يتم تنفيذها بمؤشرات معيارية «جدول المؤشرات».

٤. HOTA.EXE (C). FEB 1990 موسكو، ستاس لياشيف.

يسمح (HOTA) بالقيام بحماية البرامج من الاستخدام غير المشروع. وخلال عملية الحماية يمكن أن تكتب بعض المعلومات السرية (المفتاح) على ديسك صلب، وفي هذا الديسك (القرص) يجب القيام بتوقيف متكرر للحماية، وبعد كل عملية إعادة تنظيم الديسك. يمكن لديسك واحد متماسك إضافة أية كمية ممكنة من البرامج.

يُلحظ في البرامج المحمية إجراءات لمواجهة الفرز وذلك بمساعدة برامج مثل DEBUG, AFD, PERISCOPE يمكن هنا ترميز البرنامج المحمي. يؤخذ بعين الإعتبار حين وضع الحماية المؤشرات التالية التي قد يدخلها المستخدم:

• السمات الإفرادية للبرنامج والمستخدم: كلمة السر، اسم البرنامج، الرواية، تاريخ الصنع، تاريخ البيع، اسم المؤلف، اسم الشاري، علامات إضافية للمؤلف. هذه المعلومات تدخل في توليفة «المفتاح».

• عدد مرات التشغيل وتاريخ وضع الحماية والبرنامج المحمي غير قادر على العمل.

• معلومات: للمستخدم عن حقوق التأليف. ويمكن قراءتها بعد إدخال كلمة السر.

• الحدود: الصعوبات الممكنة في حماية الملفات المفتوحة. يمتلك نظام (HOTA) إينترفيس (قلاب) – جيد مع مشغل.

٥. PROKOP

يتألف نظام الحماية من النسخ والتقليد غير المشروع من البرامج التالية: Codprtct.com, Floprtct.com, Hardinst.com . يحدث في سياق حماية البرنامج ديسكيت مفتاحية وربط البرامج المحملة معها. والديسكيت المفتاحي لا يمكن نسخها بواسطة وسائل DOS ومثل

COPYWRIT أو COPYIIPC أيضاً. ويمكن أن يربط مع ديسكيت واحدة (12) برنامج.

عند وضع الحماية يؤخذ بعين الإعتبار المؤشرات المدخلة من قبل المستخدم:

* عدد المرات التي يصبح البرنامج المحمي بعدها غير قادر على العمل.

* عدد النسخ الممكنة في الديسك القاسي (كل النسخ التي تفوق النهاية الحدية الموضوعة لن تكون قادرة على العمل).

* سرعة الحاسوب: ويتم حسابها عند تصميم الديكسيت المفتاحي.

* الحدود: لا يمكن حماية الملفات المكشوفة. يمكن في حال إصابة الديسكيت المفتاحي بالفيروسات أن تُفقد البرامج المحمية.

٦. نظام حماية المنتجات البرامجية من النسخ غير المشروع: MAY (C),1990 الناشر «الحوار».

INST-FD.EXE	SETNU-M.EXE
INST-HD.EXE	NIN-HD.EXE

تسمح المنظومة بما يلي:

- وضع وسائل الحماية على المنتج البرنامجي المختار.
- نقل المنتج البرنامجي المحمي إلى ديسك صلب.
- تؤخذ المؤشرات التي قد يدخلها المستخدم بعين الإعتبار وهي:

* عدد النسخ الممكنة في الديسك الصلب، علماً أن جميع النسخ المصنوعة والتي تتجاوز النهاية الحدية، لن تكون قادرة على العمل كلياً أو جزئياً.

* رقم التسجيل واسم الشاري.

إذا كان مجمع البرامج يتألف من عدة نماذج، فتتم حماية البرنامج الرئيسي، والذي يزيد حجمه عن 8.5 كيلوبايت. يمكن أن يوجد في

الديسكيت منتج محمي واحد. يمكن للبرنامج المحمي أن يفقد القدرة على العمل في حال نسخه.

٧. منظومة (PP) غورلينكو، أكاديمية العلوم السوفيتية 1989:

يتألف نظام حماية البرامج من النسخ من النماذج التالية:

I-AND-R.EXE

PTOOLF.EXE

PTOOLI.EXE

تتألف الحماية من المراحل التالية:

• تغيير الملف EXE. يفصل من الملف قسم بمقياس 512 بايت، ويوضع مكانه برنامج خاص يمكنه التعرف على الشروط الموضوعة والمعطاة في لحظة

• وضع الملف EXE على ديسك صلب بمساعدة انمالياتر.

• حماية الإنستالياتر من النسخ. يجب أن تؤخذ المؤشرات التالية بعين الإعتبار حين وضع الحماية:

* عدد النسخ المكنة في الديسك الصلب. مع العلم أن كل النسخ التي تتجاوز النهاية الحدية الموضوعية لن تكون قادرة على العمل.

* رقم الحزمة – المجموعة.

٨. منظومة (TEXT PROTECTION) غورلينكو. معهد أكاديمية العلوم السوفيتية 1990. يتألف نظام حماية المعطيات الكتابية «النصية» من النسخ من النماذج التالية:

DECOD.EXE	POLD.EXE
DESHIFR.EXE	SHIFR.EXE

وتتألف الحماية من المراحل التالية:

* ترميز الملفات الكتابية بالبرنامج SHIFR.EXE.

* وضع الملفات مع المعطيات المرمزة في الديسكيت.

* إيجاد ماستر ديسكيت يسجل عليه برنامج الحماية من النسخ والنص المرمز (DECOD.EXE).

* حماية البرنامج (DECOD.EXE) من النسخ عن طريق استخدام تقنيات حماية الملفات المحملة. يمكن طباعة النص المحمي مرات محدودة.

٩. Secret Text,(c) NISS SOFT . MAR 1990.

المحتوى : Stext.exe

Svier.exe

نظام حماية ملفات النصوص من النسخ غير المشروع من القرص الصلب، يعالج البرنامج Stext.exe الملف النصي – ملف النصوص. وبعدها يمكن التعامل مع الملف من خلال البرنامج Sviewer.exe يسمح هذا النظام بما يلي:

* وضع ونزع الحماية عن النص.
* قراءة النص من الشاشة من خلال القيام بعملية «التصفح» والبحث.
* طباعة النص.

الحدود:

* لا تعالج ملفات تتجاوز 64 كيلوبايت.
* تعمل بقيادة MS-DOS الإحتمال × 3.

١٠. Program Protect. (c) NISS SOFT 3 MAR 1990 الإحتمال.

التوليفة : PPU3.COM

نظام حماية الملفات المستعملة (من نوع EXE,COM) من الإستخدام والنسخ غير المشروع مع ديسك صلب. يُرمّز الملف المحمي. تستخدم الحماية من تحليل الملف. وفقاً لرأي مؤلفي هذا النظام فإنه لنزع الحماية، يجب بذل عمل وزمن كبير لا يقل عن 720 ساعة عمل لمبرمج متوسط الكفاءة قادر على التعامل مع لغة Assembler.

يسمح هذا النظام بالأمور التالية:

• وضع الحماية من النسخ غير المشروع.

• نزع الحماية من أمام النسخ غير المشروع.

وضع حماية من النسخ غيرالمشروع دون إمكانية نزعها.

• يقلص إمكانية الوصول للملفات باستخدام عملية متابعة الرمز والتقصي عنه قبل استخدامه (يوجد عشرة رموز).

الحدود:

* لا تُعالج الملفات التي تفوق 63 كيلوبايت.

* يعمل بقيادة MS-DOS الاحتمال × 3.

١١. Secret Disk, ver 1,01. (c) 1985 by CF Syystems:

يتألف من النماذج التالية:

DESDRV.BIN	SON.COM
SECRET.BIN	SREBUILD.COM
CHANGEP.COM	SECFINIT.EXE
SOFF.COM	SECINIT.EXE
DESCOPY.COM	HIDEFLIP.COM
	SDELETE.COM

تسمح هذه المجموعة بالأمور التالية:

• نسخ ملفات Secret Disk.

• ترميز وفك رموز الملفات.

• تغيير كلمة السر.

• تشكيل Secret Disk في أي مدخل للديسك.

• استبدال الملف الرئيسي.

• إعادة تشكيل من أجل Secret Disk.

١٢. LOCK, ver 2,00 (c) 1986. Soft Logic.

Solutions Inc. Manchester N.H يتألف من النماذج الآتية:

LOCK.EXE, UNLOCK.EXE.

هذا النظام موجـه لـترميز الملفات على الديسك، وهـو سـهل جـداً في الاستخدام. ويكفي الإشارة لكلمة السر ليقوم بترميز وفك ترميز الملف.

يقوم البرنامج (Lock) بالترميز. يطلب البرنامج اسم الملف المرمـز وكلمـة السر. أما البرنامج (UNLOCK) فيفكك الترميز، ويتطلب اسم الملف وكلمـة السر.

١٣. COPY Protection, ver 2,02. (c) D.RUDEV, 1987, 1988.

يتألف النظام من النماذج التالية:

PROTECT.INS PROTECT,COM

PROTECT.LOD

موجه ومخصص لإلحاق وربط البرنامج مع الديسـك المفتـاحي والديسـك الصلب. تسمح هذه المجموعة بما يلي:

- كتابة ومحي مفتاح حماية الديسكيت.
- حماية الديسكيت من العرض والملاحظة.
- القيام بالملاحظة والإستبدال.
- تسجيل البرنامج على
- إضافة اسم البرنامج من قائمة الأقسام.
- تسجيل البرنامج مع نص الحماية على الديسكيت.
- التسجيل على القرص الصلب مع نص حماية البرنامج.

١٤. ANTY COP, (c) كييف 1989.

توليفة النظام:

ANTY.COP ANTYCOP.ZAGANTY.UST

وهو عبارة عن نظـام لحمايـة الملفـات الموجـودة في القـرص الصلـب مـن النسخ غيرالمشروع. تضم الملفات المحمية إلى ناقل وحين النسخ تصبـح غـير قادرة على العمل، يمكن من خلال هذا النظام القيام بما يلي:

- وضع ونزع حماية الملفات.
- نقل ونزع حماية الملف.
- نقل البرنامج المحمي إلى فينتشيستر.
- تغيير مفتاح الحماية المستخدم للترميز.
- الحصول على معلومات عن الملف المحمي.
- يجب الأخذ بعين الإعتبار حين وضع الحماية الأمور التالية:

* رقم النسخ المكن في القرص الصلب. ذلك لأن جميـع النسـخ الـتي تتجاوز كميتها القيمة العظمى المكنة، والموضوعـة والمحـددة سـتكون غـير قادرة على العمل.

* مفتاح الحماية.

* معلومات حول حقوق المؤلف.

١٥. SHIELD.EXE, (c) ALEX SOFTWARE, 1989:

يسـمح بتكويـن برامـج محميـة مـن خـلال تكويـن ديسـكيت مفتـاحي. وتؤخذ المؤشرات التالية بعين الإعتبار حين وضع الحماية:

- عدد الأعظمي في القرص الصلـب. النسـخ الـتي تفـوق كميـة النسـخ العظمى المحددة لن تكون قادرة على العمل.
- رقم المجموعة.

١٦. PROTECT-5, (c), 1990 شيرباكوف:

يتألف النظام من:

SRTPT.EXE	INIPTE.EXE
MYFOR.COM	SETPTK.EXE

RUNM.COM PROTECT.EXE

يسمح بحماية ملفات النص، قواعد المعطيات، ملفات EXE. COM المستخدمة من النسخ غير المشروع، ويمكن استخدام كلمة لتحجيم إمكانية الوصول للنماذج المحمية.

مبدأ العمل:

تحويل المعطيات لكود مناسب مع الأخذ بعين الإعتبار خصائص الحاسوب الإلكتروني وخاصة الديسك الصلب، فيديو أدابتر، البوابات. عدا عن ذلك يتم أيضاً في سبيل الحماية من النسخ غير المشروع استخدام الرابطة و الوصل مع الديسك المجهزة لتكون مفتاحية. وفي هذا السياق يتم استخدام السكة ذات الرقم (2a) بعد، وعلى هذه السكة في القطاع الأول، يتم تسجيل المعلومات المفتاحية من نوع:

6566, 6a64, 6a74, 3076, 0030, 0009, 090, 0......

١٧. حماية ميكرو. إينتربروغرام ١٩٨٧:

التركيب:

SCUR.EXE

$SPRO-PE_2$ .EXE

$SPRO-PE_3$ .EXE

$SPRO-PE_4$ .EXE

الوظائف:

• الحماية من النسخ والتوزيع غير المشروع المنتجات البرامجية، والمسجلة على نواقل مغناطيسية، باستخدام ديسكيت خاصة بذلك.

• الحماية لتأكيد حقوق استخدام المنتج البرامجي.

• الحماية لتأكيد حقوق المؤلف.

• الحماية لتحديد زمن وعدد مرات استخدام المنتج.

هذا النظام يعمل على أجهزة من طراز IBM PC/XT.

أو المشتركة معها تحت قيادة المنظومة العملياتيـة MS-DOS والاحتمـال 3.10 فما فوق.

١٨. MAWR, V 5.01.(C) N-,1989:

التركيب:

ASPYRIN.EXE

MAVR.EXE

PERS.COM

يسمح نظام الحماية بكلمة السر القيام بالأمور التالية:

- الحماية الكاملة للقرص الصلب من محاولات الوصـول غـير المشـروعة إليه.
- حماية أقسام ملفـات المنظومـة MS-DOS مـن الوصـول غـير المشـروع إليـها.
- مراقبة نسخ المعلومات من القرص الصلب إلى قرص فلوبي ولدى تقـدم هذه المنظومة إمكانية منع العمل تماماً مع هذه الأقراص.
- مراقبة تغيير ملفات SYS, COM, EXE:

ومـن أجـل حمايـة الديسـك يوضـع ديسك خـاص لكلمـة السـر. وتتـم الحماية بهذه الطريقة من خلال إيجاد هيكلية FAT غير معياريـة، ولذلـك وحـين التحميـل مـع أقـراص فلوبـي لا يمكـن الوصـول بعدهـا والنفـاذ إلى الديسك الصلب.

١٩. PROTECT.COM, V 3.0:

غورتيناك (C), 90 . 4.12

تسمح منظومة PROTECT بوضع الحمايـة مـن التسـجيل أو القـراءة في أقسام المجزء المتوضع في أحد الـبرامج التاليـة: FDISK, DISK Manager ولايمكن النفاذ للأقسام إلا عبر كلمة السر فقط. والفيروسـات الموجـودة غـير قـادرة عمليـاً علـى اخـتراق هـذه المنظومـة، ويمكـن القيـام بالحمايـة مـن الحمولة.

٢٠. UNLOCK, V2.0

(C) 1988, NU-GMbH

للحماية من النفاذ الى القرص الصلب. والعبور لايتم إلا بكلمة السر. في حال التحميل مع الديسكيت فإن «الفينتشستر» غير مرئي لأوامر DOS.

٢١. LIST, (C) 1991, V 1.0:

التركيب: LIST.EXE

وهو عبارة عن نظام تحديد وتقليص إمكانية الوصول والحماية من النسخ ونسخ الملفات على القرص. يتم تسجيل الملفات في القائمة «السوداء» مما يمنع إمكانية نسخها أو عزلها أو تغييرها. يمكن إسترجاع الملف من القائمة «السوداء».

حين أوردنا بإيجاز الأنظمة الحمائية الموجودة لم نتعرض وعن وعي لطرائق نزع وكشف تلك الحمايات لأسباب أدبية، وكي لا نزيد من عدد الخصوم لنا في أوساط مصممي ومستخدمي تلك الأنظمة، ولإعطاء فرصة لهواة التنقيب في البرامج المحمية للاستمتاع.

وبعد هذا العرض سينشأ لدى القارئ. إنطباع حول صعوبة التعامل مع هذه الأنظمة وآليات الحماية التي تفرضها.

نورد فيما يلي الإتجاهات الأساسية في فن «تجريد» البرامج المحمية.

الفصل الرابع

كيف تجرد برنامج

لننظر بإيجاز شديد في مسألة كيفية تجنب وسائل الحماية القائمة، بما فيها التي أوردناها في الفصل السابق؛ عملياً لايوجد وضع لايمكن الخروج منه بل يوجد طريق خاطئ للخروج. حين تحدثنا عن وسائل الحماية تطرقنا جزئياً لمسائل نزع الحماية أيضاً. سنتوقف هنا عند ثلاثة محاور:

١. الوسائل المعيارية.

٢. المعدات الخاصة.

٣. مبادئ وضع «مجرد» عام وعمومي شامل.

4.1 بعض الوسائل المعيارية لنزع الحماية:

سننظر هنا في بعض البرامج واسعة الإنتشار، والمستخدمة غالباً لتحليل ونزع حماية برامج أخرى. لم تكتمل قائمة تلك البرامج، وهي تؤشر فقط الى الإتجاهات الرئيسية لتطورها. يسمح الإستخدام المباشر للبرامج الموصوفة أدناه بنزع الحمايات البسيطة نسبياً. أما استخدامها كوسيلة إضافية مع برامج متخصصة فإنه يسمح بتحقيق نجاحات جيدة في هذا المجال.

إضافة لذلك فإن الوسائل المعيارية تتطور وتكتمل دائماً ودون إنقطاع، وما هو غير ممكن اليوم، يصبح غداً أمراً عادياً. ومن الممتع ملاحظة أن بعض البرامج المتخصصة والموجهة أساساً لنزع الحمايات تحوي تحذيراً من مخالفة قانون حماية حقوق المؤلف التي قد تنشأ نتيجة لإستخدامها.

4.1.1 بعض التجهيزات عمومية الوظائف:

وهي تسمح بالقراءة والتحليل وتسجيل مختلف المعلومات علـى القـرص وفي الذاكرة العملياتية. وهذه المجموعة قائمة في العديد من البرامج، ونلحـظ منها التالية:

•مجموعة نورتون:

مجموعة برامج (C) Petre Norton

PC-SHELL, (C) Central Point Soft ware Inc

•ملاحظة وتسجيل مقاطع مفردة من الديسك.

•ملاحظة الذاكرة العملياتية.

•إمكانيات إضافية كثيرة.

DISK Explorer, July 1985, (c) Quaid Software Ltd.

ملاحظة وتسجيل مقاطع مفردة من الديسك في أي شكل وإطار كان.

MAPMEM.COM. Borland Inc.1986.

Mpram.EXE, CSPI, 1984.

FX-SHOW.COM V,1.53.

KONONENKO ,1991.

PEEK.COM, V,1.0.

مـن خـلال البرنـامجين الأولـين يمكـن الملاحظـة فقـط. بينمـا يسـمح البرنامجان الآخران بالملاحظة والتدقيق والتنقيح في الذاكرة العملياتية.

4.1.2 المنظم والمفكك.

نلحظ الآتي منها:

•Quaid Analyzer, (c) Quaid Software Ltd, may 1988.

•Periscope III, (c) Data Base Decisions, v, 4, 1990.

•Turbo Debugger, (c) Borland Inc, v2. 01, 1990.

•Souser, (c) V Communication, v. 1.72, 1988.

•Disdos, (c) RJSwantek. Inc,v.2.3, 1988.

4.1.3 البرامج الموجهة لنزع الحماية:

برامج هذه المجموعة موجهة، خاصة لنسخ النواقل المحمية وكشف بعض أنواع الحمايات.

• مجموعة Copy II-PC v, 5.01 , 1988, (c) Central Point Software, Inc.

وتتألف من البرامج التالية:

Nokey.com, Fixup.com, Noguard.com, Bulkeras.exe, CopyII Pc.exe

ويمكن من خلالها القيام :

* بنسخ الأقراص المحمية.

* بنسخ النماذج المحمية بمجموعات SIERRA, EVERLOCK, PROLOK, SUPERLOK.

* بنقل المجموعات المحمية إلى .

* بتحقيق إحتمالات البرنامج COPY II PC.EXE والتي تسمح بنسخ الأقراص لأجل الحاسوب من طراز: Apple Makintosh, Commodore 64/128 وهذه الشركة تنتج COPYIIPC OPTION BOARD خاص، يقوم بمشاركة البرامج المطابقة بعملية تبديل أعظمية للأقراص .

• برنامج CopyWrite Apr 1980 روبرت ماكوايد وفرانك كلين، سوفت وير:

وهو مخصص لنسخ الأقراص المحمية. ويقوم تحديداً بنسخ أقراص مجموعة EverLock , Prolok.

• Unlock.com Transec Master Key/ver c-3.31 :

لنسخ نماذج محمية بواسطة المجموعات Prolok, Everlock, Sierra.

4.2 المعدات الخاصة:

بداية لابد من القول أنه غالباً يمكن نزع الحماية عن المجموعة التي تستخدم أساليب بدائية وعلى قاعدة الوسائل المعيارية. خاصة عندما يقوم هاوٍ بوضع الحماية، ويستخدم فيها آليات حماية مبسطة. في هذه الحالة ينصح بتجريب ما يلي:

- النسخ الكامل للفينتشستر من خلال الشريط المغناطيسي.
- الإنتباه ومتابعة الإستخدامات للـ (13h) (21h) وذلك بمساعدة برنامج ذو مقر غير كبير.

ونورد أدناه مثالاً له أن تحجز في الملف EXE الإنتقالات والمواقع المشكوك فيها بأوامر (goh) NOP.

- الإقلاع والتشغيل مع أي من المنظمات.
- «معالجة» القرص المحمي بمساعدة برامج مماثلة لـ NDD... الخ.

تقول حكمة شرقية «عندما يلقي الحداد بنفسه على العربة، يعتقد الجميع بأنه سيقع أرضاً، ولا يفكر بأن العربة ستقع وليس هو. إنه لأمر مضحك» وفي مثال «تجريد» البرنامج فإن العربة تغير إتجاهها أيضاً. ثم وبمهارة كافية صنع العديد من المجموعات وخاصة Copylock و Hota. ويحتاج من يريد تفحص مبدأ عملها وفهمه لكثير من الوقت. فإذا كان الوقت غير كاف ومبدأ الحماية قائم على أُسس غير معيارية لكنه يُمكّن من الوصول للمعطيات المسجلة على الشريط المغناطيسي، فإنه من المستطاع القيام بمحاولة للحصول على نسخة دقيقة للديسكيت المفتاحي، بمتابعة كيف تؤطر المجموعة ذاتها – من خلال النماذج القائمة – وتكتب وتقرأ الديسكيت المفتاحي.

نورد أدناه نص البرنامج المخصص لمتابعة عمل المسألة المبحوثة. يتوضع هذا البرنامج في مجال DOS (في مجال معالجة 13h) ويظهر على الشاشة معلومات نمطية حول محتويات Register ومجالات الذاكرة. يسمى البرنامج «تيلي كاميرا». مبدئياً وقبل التشغيل، على المستخدم أن يختار

مكان في مجال الذاكرة، حيث يريد هو أن توجد «التيلي كاميرا» العائدة له.يتم الإختيار بمساعدة أيٍ من المنظمات.

يؤخذ عنوان وجهات معالجة الإنقطاعات الضرورية مباشرة من الذاكرة العملياتية ومن مجال ساحة الكميات الموجهة. بعد ذلك وحسب العناوين المختارة يتم تفكيك مجال الذاكرة المطابق وتخزن عناوين الأمرين المتتاليين المنفذين إضافة للأوامر ذاتها.

يتم تنفيذ العملية بيسر وسهولة من خلال Turbo Debuger المعروف جيداً. ويتم ترميز العناوين المختارة والأوامر على شاكلة العنوان 1, العنوان 2, الأوامر 1, الأوامر 2.

إختيار أمرين يمكن تفسيره بضرورة استرجاع الأمر الأولي واسترجاع نقطة الإنقطاع للإنتقال اللاحق للعملية الحسابية في مقطع الذاكرة المحدد. يعمل البرنامج وفق اللوغاريتم التالي:

الخطوة ١:

العمليات التمهيدية – لأولية: تكتب أوامر استخلاص الإنقطاعات وفقاً للعناوين المختارة، مثلاً int 56h, int5ah ومن ثم تسجل عناوين ملاحق البرنامج المعالجة لمعطيات الإنقطاع الخروج ولكن يتبقى المقر.

الخطوة ٢:

في الإنقطاع الأول تنفذ العمليات التالية:

- حفظ Register.
- الإيضاح على الشاشة محتوى Register ومجالات الذاكرة العملياتية الضرورية.
- استرجاع الأمر 1 بالعنوان 1.
- وضع أمر تقديم وإعطاء الإنقطاع بالعنوان 2.
- تقليص المحتوى الموجود في IP Register إلى 2 من أجل ألا تنتقل القيادة في نهايته معالجة الإنقطاع نحو الأمر التالي، ولكن للاسترجاع حسب العنوان 1 ووفقاً للأمر 1.
- إعادة إرجاع Register والخروج من الإنقطاعات.

الخطوة ٣:

في الإنقطاع الثاني تنفذ العمليات التالية:

- حفظ Register وتذكرها.
- إرجاع الأمر 2 حسب العنوان 2.
- إرجاع أمر نقل الإنقطاع حسب العنوان 1.
- تقليص المحتوى الموجود في IP Register إلى 2 من أجل ألا ينتقل التحكم في نهاية معالجة الإنقطاع نحو الأمر التالي. ولكن لإرجاع ذلك إلى العنوان 2 ووفقاً للأمر 2.
- إرجاع Register والخروج من الإنقطاع.

ملاحظة: لاتنسوا بأن برنامجكم ذو مقر دائم؛ ولذلك ولأجل تجنب مشاكل صغيرة غير مريحة، يمنع استخدام انقطاعات DOS.

المثال 4.1:

```
sseg segment stack
        db 256 dup(?)
sseg ends
dseg segment
dseg ends
cseg segment
        assume cs:cseg,ss:sseg,ds;dseg
z.      proc far
        push ds
        mov ax,0
        push ax
```

تحضير DOS للعمل مع البرنامج:

: في مجال DOS المسؤول عن معالجة (13h).

تُكتب أوامر استدعاء المسألة int 5ah معالجة 5ah تتم من قبل :

- ريجستر Segment عنوان معالجة (13h) :mov bx, 70h:

```
oush bx
pop ds
mov bx, of 86h
```

Register تداخل نسبة :(segment)

```
mov ax, 5a cdh
mov ds:[bx[,ax
```

توجيه البرنامج لمعالجة الإنقطاعات، الأولى منها:

```
push cs
pop ds
mov dx,offset prod
mov ah,25h
mov al,5ah
int 21h
```

الثانية :

```
mov dx,offset prod1
mov ah,25h
mov al,5bh
int 21h
```

الخروج والبقاء في المقر:

```
mov dx,100h
mov al,0
mov ah,31h
int 21h
```

برنامج معالجة الأولى:

```
prod: push ds  : +12
push bx           : +10
push ax           : +8
push dp           : +6
push dx           : +4
push es           : +2
push cx           : +0
```

تجهيز الشاشة لتقديم المعلومات حول محتوى Register

```
mov al,1
vad:mov ah,5h
int 10h
mov dl,0dh
mov ah,2h
int 21h
mov dl,0ah
mov ah,2h
int 21h
:____________
mov dp,sp
```

```
mov ax,[bp+14]          ; Ip Register: طباعة المحتوى
cal dec1
mov ax,[bp+16]          ;cs
call dec1               ;
mov ax,[dp+8]           ;ax
call dec1               ;
mov ax,[dp+4]           ; dx
call des1
```

الإنتقال لسطر جديد:

```
mov dl,0dh
mov ah,2h
int 21h
mov dl,0ah
mov ah,2h
int 21h
```

طباعة قائمة مؤشرات الديسكيت:

```
mov bx,0
push bx
pop ds
mov bx,522h
mov ax,ds:[bx]
call dec1
add bx,2
mov ax,ds:[bx]
```

```
c3ll dec1
add bx,2
mov ax,ds:[bx]
call dec1
add bx,2
mov ax,ds:[bx]
call dec1
add bx,2
mov ax,ds:[bx]
call dec1
add bx,2
mov ax,ds:[bx]
call dec1
```

الإنتقال لسطر جديد :

```
mov dl,0dh
mov ah,2h
int 21h
mov dl,0ah
mov ah,2h
int21h
```

طباعة المقاطع : وتستخدم فى حال عملية التأطير

```
mov bx, [bp+10]
mov ax,es:[bx]
call dec1
```

```
mov ax,es;[bx+2]
call dec1
mov ax,es:[bx+4]
call dec1
mov ax,es:[bx+6]
call dec1
mov ax,es:[bx+8]
call dec1
mov ax.es:[bx+10]
call dec1
mov ax,es:[bx+12]
call dec1
mov ax,es:[bx+14]
call dec1
mov ax,es:[bx+16]
call dec1
mov ax,es:[bx+18]
call dec1
mov ax,es:[bx+20]
call dec1
```

إيقاف أمر نقل الإنقطاع بالعنوان 2

```
mov bx,70h
```

ريجستر segment: لعنوان معالجة 13h:

```
push bx
```

```
pop ds
mov bx,0f88h :segment التداخل نسبة لريجستر
mov ax,5bcdh
mov ds:[bx],ax
```

إرجاع الأمر فى العنوان1

```
mov bp,sp
mov bx,[bp+14];lp
mov ds,[bp+16];cs       حيث وصلنا لـ DOS
sub bx,2
mov ax,9090h            الأمر الأول
mov ds;[bx],ax          ;ds:[bx]
mov [bp+14],bx
```

إعادة الشاشة لوضعها الأولى:

```
mov dl,0dh
mov ah,2h
int 21h
mov dl,0dh
mov ah,2h
int 21h
sc: mov ah,0h
int 16h
mov ah,5h
mov al,0
int 10h
```

```
aa:  pop cx
     pop es
     pop dx
     popbp
     pop ax
     pop bx
     pop ds
     iret
```

برنامج معالجة الثانية :

```
prod1:push ds
      push bx
      push ax
      push bp
      push dx
      push
      es
      push cx

      mov bx,70h

      push bx
      pop ds
      mov bx,0f86h
```

```
mov ax,5acdh
mov ds:[bx],ax


mov bp,sp
mov bx,[bp+14]   ;ip
mov ds,[bp+16]   ;cs
sub bx,2
mov ax,9090h
mov ds:[bx],ax     ;ds:[bx]
mov[bp+14],bx
push cs
pop ds
mov dx,offset prod
mov ah,25h
mov al,18h
int21h
pop cx
pop es
po dx
pop bp
pop ax
pop bp
pop ds
iret
```

```
cseg ends

end z
```

من الضروري ملاحظة أنه في المثال السابق، ولأجل سحب التحكم من مجال DOS فقد استخدمت أوامر لإستدعاء الإنقطاعات في مجال الذاكرة. ولكن ذلك ليس هو الطريقة المثلى والوحيدة.

وعوضاً عن الأوامر من طراز int يمكن استخدام أوامر jmp , call والهام في الأمر عدم نسيان تنفيذ الأوامر الأولية لـ DOS المستبدلة بها. وفي حال إذا كانت المسألة المحمية والمبحوثة – المتابعة تعمل بصورة مباشرة مع BOIS متجنبة DOS فإن الوسيلة الوحيدة لمتابعتها، تكمن في إجراء تبديل (موديفيكشن) في BOIS.

وبنفس المبدأ يمكن أيضاً توضع «تيلي كاميرا» وبحذر في BOIS. عدا ذلك فإن إيجاد وسائل ومعدات خاصة، يعتبر أمراً هاماً، ومساعداً لنزع الحماية. وتلك المعدات تقوم بالتحكم بعمل الحاسوب، ونتائج عمل البرامج التي لايمكن إطلاقها بالطرق التقليدية. ويمكن القول، بأن الخطوة التالية لتطور أنظمة نزع الحماية، إنما تكمن في الإنتشار الواسع لـ «ريترانسلياتر» الذي يسمح بإسترجاع البرنامج المكتوب، وفي حال تداخلها أيضاً.

4.3 «المجرد» النازع العام

4.3.1 مبادىء البناء الأساسية:

تكمن المبادىء الأساسية لبناء «المجرد» العام، والذي يعتبر تجسيداً لأحد الطرائق الديناميكية لنزع الحماية فيما يلي:

١. يشكل «المجرد» في (MD) ملفاً يعكس بصورة كاملة محتوى الذاكرة العملياتية.

لأجل ذلك، يتم نسخ كامل الذاكرة العملياتية، بما فيها مساحة ومكان كميات التوجه في DOS.

٢. يضع «المجرد» البرنامج المبحوث فيه، في ملف في ذلك المكان والشكل، كما هي عليه في الذاكرة وكما كانت قبل تنفيذ الأمر الأول.

٣.يقوم «المجرد» بالانتقاء من الملف وذلك بمعدل أمر واحد كل مرة، ويقلد عملية تنفيذه. ويعلم «المجرد» المستخدم عن ذلك ويستدعي إذا تطلب الأمر «كود» الإنهاء لهذا الأمر أو ذاك، ويسمح «المجرد» بملاحقة كل التوجهات ضمن نطاق المسألة في الملفات، إلى التجهيزات الخارجية. ونحو كميات التوجه.

٤.يقوم بصورة «ثانوية» بتفكيك البرنامج.

يسمح للمستخدم أن يتعامل معه كما مع المنظم الطبيعي. وفي هذه الحالة فإن جميع طرائق الحماية (الفصل الثاني) تصبح غير فعالة ودون نتيجة. و«المجرد» من حيث المحتوى لايقوم بعمل المعالج «بروسيسور» فقط بل المنظومة العملياتية وكامل وظائف DOS وBOIS. ويعتبر وسيلة عامة للصراع ضد الفيروس أيضاً. وذلك لأنه في مجرى عملية تنفيذ البرنامج يسمح بمتابعة ورصد جميع أعماله، وعملياته غير الجيدة، وتمييز وتحديد أجزاء البرنامج المشكوك بها أو الخاضعة للشك. وقد أطلق المصممون تسمية تجارية على هذا النوع النازع للحماية وهي SREDA.

اعتبر الهدف من تصميم «المجرد» العام في إيجاد مجموعة برامج لتحليل «كود المسألة المنفذة والمستخدمة للعمل مع الأجهزة الحاسبة الألكترونية من طراز IBM PC في وسط DOS.

هدف المجموعة:

تكمن مهمة هذه المجموعة في :

- أبعاد أية برامج لاتحوي نصوص أولية.
- إختبار البرامج في مسألة تبيان وإستبعاد الفيروس، وآليات الحماية من النسخ غير المشروع.
- إختبار عمل DOS وBOIS.

4.3.2 توليفة المجموعة:

يدخل في توليفة المجموعة النماذج المحققة للوظائف التالية:

- تفكيك الكود المنفذ.
- تنفيذ الكود المستخدم و.
- التنفيذ وتقديم معطيات حول وظائف DOS.
- خدمات تنظيم.

4.3.3 نظام العمل:

يبدأ العمل مع المجموعة من إطلاق برنامج SREDA ومن ثم يمكن إنجاز العمل لاحقاً من خلال المواقف الأساسية التالية:

- تحميل البرنامج المختبر (بما فيه النماذج مع التوسيع ,BIN, COM EXE إضافة إلى الربط والحبكة مع العام وبدءاً من العناوين المحددة).
- إظهار البرنامج المختبر وأقسامه والوسط الذي يجب أن يُنجز فيه على الشاشة وفي الملف على شكل ASCII كود ستة عشري وعشري، وكنص مفكك.
- تنفيذ البرنامج المحمل بأي عنوان وبأية شروط أولية.
- إعادة «الكود» المطلوب للمسألة المختبرة (والتي يحددها المستخدم) لحالة الإنهاء. بالنسبة للأوامر المحددة. ووظائف DOS.
- الحصول على كامل المطلوب تنفيذ البرنامج، وكذلك أنماط الأوامر الإفرادية المفرزة. مثل أوامر out, in, cme, int.
- الإعادة وفقاً لأوامر تنفيذ البرنامج المحمل مع الملاحظات حول الغاية منه و «الكود» الممكن للأمر المنفذ في اللحظة ذاتها المحددة أو لوظائف DOS.
- تنفيذ البرنامج المحمل بسرعات مختلفة مع عكس سياق العمل.
- السماح للانقطاعات العائدة للعملية بأن تنفذ في أي لحظة.
- وضع العلامات والقياس لأجل التبقي.
- وفقاً للعناوين المختارة.
- وفقاً لقيمة ومغزى Register.

- حسب نوع الأمر.
- وفقاً للشروط المنطقية الموحدة للمعلومات المذكورة أعلاه.
- معالجة وتدقيق وإختبار المسألة المبحوثة على مدى وجود الأنواع المعروفة من الفيروسات فيها، وكذلك آليات الحماية من النسخ غير المشروع.

4.3.4 المتطلبات نحو «المجرد»:

١. يجب أن تضمن المجموعة تحميل المسألة وإختبارها، وبغض النظر عن الحجم الذي تشغله في الذاكرة مع أية آليات حمائية كانت سواء من المنظم أو من المفكك.

٢. يجب ألا تسمح المجموعة بتحميل الكود الخاص بها المنفذ سواء كان الأصل أم نسخة عنه وكأنه هو المسألة المحملة الخاضعة للإختبار. أي ألا تقوم باختبار ذاتها. ويمكن ملاحظة النقائص الأساسية التالية للمجرد:

- صعوبة ايموليشن للوظائف غير الموثقة – غير الوثائقية لـ DOS.
- صعوبة ايموليشن ديناميكية عمل التجهيزات.

الخاتمـــة

إذا نشـأت لديكـم صعوبـات لايمكـن تجاوزهـا أثنـاء تصميـم منظومـات الحماية أو نزعها، فحاولوا أن تضعوا أنفسكم في مكان ذلك الشخص الـذي يقوم بعملية معاكسة ووظائف مضادة.

وواضع الحماية يجب أن يتصور نفسه مكان نازعها وبالعكس. ومُصنّـع السم يتابع واضع الترياق وبالعكس.. الخ.

ما زالت مسألة وضع الحماية ونزعها بعيـدة تمامـاً عـن حلهـا النهـائي وتشكل حقلاً واسعاً للعمل وتحدياً للدماغ والأبحـاث في هـذا المجـال تقـود دائماً إلى ظهور أسئلة جديدة تتطلب الإجابة.

ختاماً نتمنى لكم النجاح.

ثبت المراجع

١. هوفمان، بوسو. «القرصنة في قرن الحاسوب» ١٩٩٠. دار نشر «مير».

٢. شارب، «الإستخدام غير القانوني للوسائل البرنامجية في بريطانيا العظمى»، «الكفاح لم ينتهي» ١٩٩٠ ، دار نشر «مير».

٣. دورسيليرفان، لاو «أمام أوروبا طريق واحد ضروري لحماية الوسائل البرنامجية وحقوق التأليف». دار نشر «مير» ١٩٩٠ رقم المجلد ١.

٤. سبينسر. «حماية المعلومات» دار نشر مير ١٩٩٠ المجلد ٢.

٥. نافوزنيكا، «حول الأخلاق في حق المؤلف» ١٩٩٠ دار نشر مير المجلد ٢.

أفاناسييف. مانايف، «إمكانية حماية المعلومات حين معالجتها». دار مير المجلد١٩٩٠.

دافيدوفسكي، ماكسيوف. «مدخل لحماية المعلومات»، إينتر كمبيوتر ١٩٩٠ رقم ٣.

٦. أيرامسون، «استيراد منتجات البرامج» (البداية) ١٩٩٠.

٧. غالوبيف، دوبوف، بافلوف، «جرائم الحاسوب وحماية المعلومات في الأنظمة الحاسوبية» سلسلة : الجديد في الحياة، العلم والتكنيك ١٩٩٠ دار المعرفة.

٨. سميد، برانستيد، «معيار ترميز المعطيات الماضي والمستقبل» ١٩٨٨ الرقم (5) المجلد 76.

٩. غيراسيمينكا، رازماخنين، رودينوف، «الحماية التكينكية للمعلومات» ١٩٨٩ العدد 12 «علم الراديو إليكترون الخارجي».

١٠. بلوخينتسين، كوخمانيوك، سفيريدون. «مرة أخرى حول حماية المعلومات في IBM PC». مجلة «سوفت بانوراما» ١٩٩٠ العدد (10).

١١. سياو، كيب، ميدنيك، «حماية الآلة الحاسبة الإلكترونية» دار نشر مير ١٩٨٢.

١٢. هوفمان. «الطرائق الحديثة لحماية المعلومات» مجلة الراديو السوفيتي ١٩٨٨.

١٣. شنيدر. «لغة Assembler من أجل الحاسوب الشخصي لشركة IBM» دار مير ١٩٨٨.

١٤. نورتون «التجسيد البرنامجي الجهازي للحاسوب IBM PC».

١٥. جوردين «دليل المبرمج للحاسوب الشخصي شركة IBM».

١٦. ماركفارت «من المجتمع الصناعي إلى المعلوماتي» إينتر كمبيوتر ١٩٩٠ عدد ٥.

الفهرس

حماية ونزع حماية البرامج «منظومة MS DOS»/
تـأليف مجموعـة مؤلفـين؛ ترجمـة فـايز البرشـــة؛ مراجعـــة
مؤنس كيالي. ـ دمشق: دار الطليعة الجديدة، ١٩٩٦. ـ
١ ـ ٠٠١,٦٤ ب ر ش ح ٢ ـ العنوان ٣ ـ البرشة
ع ـ ١٩٩٦/٧/ ٨٠٩
مكتبة الأسد

من المفيد جداً التذكير بالتجربة الأجنبية التي تحولت فيها السرقة إلى صناعة وليست شكلاً للتفكير. في الأعم طريقة حتى الدرجة العالية بحيث أن من جهة تقوم مؤسسات ومنظمات ومختصون خصوصاً بإنتاج وسائل حماية منظومات الحاسب وحماية المعلومات. لأنهم يعانون من خسائر محسوسة في مجال تصريف منتجاتهم لعدم وجود سوق ثابتة ومستقرة. ونقص الدعاية وشبكة الخدمة.. الخ.

ومن جهة ثانية فإن عدداً متزايداً من مستخدمي الحاسب يعانون من حاجة ملحة لحماية المعلومات التي لديهم أو يعتقدون بضرورة تأمين تلك الحماية ولكنهم لا يدرون إلى من يجب التوجه لتأمين المساعدة المختصة للوصول إلى النقاط التي ليس لدى الكثيرين منهم أي تصور حول جوهر المسألة والتقنيات المستخدمة للحماية اليوم وهذا ليس لنقصهم بالفهم.. بل لمشاغلهم.

يقدم هذا الكتاب الإجابة عن كيفية حماية البرامج ولزوم هذه الحماية.